



UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS

FACULTAD DE INGENIERÍA DE SISTEMAS E INFORMÁTICA

E.A.P. DE INGENIERÍA DE SISTEMAS

**Integración de los sistemas informáticos de la empresa
ADALTEX utilizando Oracle Service Bus como
middleware basado en el estándar SOA**

Tesina

Para optar el Título de Ingeniero de Sistemas

AUTORES

Karin Zaida Acero Navarrete

Danny Humberto Escalante Gallardo

LIMA – PERÚ
2012

FICHA CATALOGRÁFICA

ACERO NAVARRETE, Karin Zaida

ESCALANTE GALLARDO, Danny Humberto

TITULO DE LA TESINA: INTEGRACIÓN DE LOS SISTEMAS
INFORMÁTICOS DE LA EMPRESA ADALTEX UTILIZANDO
ORACLE SERVICE BUS COMO MIDDLEWARE BASADO EN EL
ESTANDAR SOA

APLICACIÓN DE SOFTWARE
(Lima, Perú 2012)

Tesina, Facultad de Ingeniería de Sistemas e Informática,
Pregrado, Universidad Nacional Mayor De San Marcos

Páginas 112

DEDICATORIA:

Dedicado a nuestros padres por todo el esfuerzo y confianza puesto en nosotros, y a nuestros profesores por las enseñanzas y apoyo brindado durante nuestra formación profesional.

UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS

Universidad del Perú, DECANA DE AMÉRICA

FACULTAD DE INGENIERIA DE SISTEMAS E INFORMÁTICA

ESCUELA ACADEMICO PROFESIONAL DE INGENIERÍA DE SISTEMAS

INTEGRACIÓN DE LOS SISTEMAS INFORMÁTICOS DE LA EMPRESA ADALTEX UTILIZANDO ORACLE SERVICE BUS COMO MIDDLEWARE BASADO EN EL ESTANDAR SOA

Autor: Acero Navarrete, Karin Zaida
Escalante Gallardo, Danny Humberto

Título: Tesina, para optar el Título Profesional de Ingeniero de Sistemas

Asesor: Walter Contreras

Fecha: Agosto del 2012

RESUMEN

Durante varias generaciones, los sistemas de las empresas han servido para un propósito específico a un único usuario o grupo de usuarios, los cuales actúan como la interfaz de dicho sistema con el resto de la organización, con lo cual se ha limitado su conexión con otros sistemas modernos o más amplios en la empresa. La integración de aplicaciones y procesos de negocio es una prioridad para la gran mayoría de las empresas de hoy en día. Uno de los retos que encaran las organizaciones modernas es dar a sus empleados información completa en tiempo real. Las aplicaciones en uso actualmente se apoyan en tecnologías antiguas, por lo cual los sistemas enfrentan dificultades a la hora de transmitir información entre las aplicaciones.

Lo que buscamos en esta tesina es proponer una solución efectiva para resolver muchos de estos problemas mediante la integración de sistemas internos en la empresa ADALTEX con una arquitectura flexible y adaptable a las tecnologías utilizadas en el desarrollo de sistemas informáticos. Logrando así un mecanismo para incrementar el conocimiento de la organización y crear ventajas competitivas futuras a la empresa.

PALABRAS CLAVES

- Bus de Integración, Integración de sistemas empresariales, SOA, Patrones de Integración, JMS.

MAJOR NATIONAL UNIVERSITY OF SAN MARCOS

University of Peru, dean of America

FACULTY OF SYSTEMS ENGINEERING AND INFORMATICS

Academic Professional School of Systems Engineering

INTEGRATION OF ENTERPRISE SYSTEMS USING ORACLE SERVICE BUS ADALTEX AS BASED ON THE STANDARD MIDDLEWARE SOA

Author: Acero Navarrete, Karin Zaida

Escalante Gallardo, Danny Humberto

Adviser: Walter Contreras

Title: Thesis, to choose the Professional Title System Engineer

Date: 2012, August

ABSTRACT

For generations, the systems of the companies they serve a specific purpose to a single user or user group, which act as the interface of the system with the rest of the organization, which has limited its connection with other modern or larger systems in the company. Integrating applications and business processes is a priority for the vast majority of companies today. One of the challenges facing modern organizations is to give employees comprehensive information in real time. Applications currently in use rely on old technology, so the systems are facing difficulties to communicate information between applications.

What we seek in this thesis is to propose an effective solution to solve many of these problems by integrating internal systems in the enterprise architecture ADALTEX with a flexible and adaptable to the technologies used in developing computer systems. Achieving a mechanism to increase awareness of the organization and create future competitive advantage to the company.

KEYWORDS

- Bus Integration Enterprise Systems Integration, SOA, Integration Patterns, JMS.

INDICE

Resumen	5
Palabras Claves	5
Abstract	6
KeyWords	7
Capítulo I	
1. Introducción.....	11
1.1 Antecedentes.....	12
1.2 Definición Problemática	13
1.3 Formulación del Problema	15
1.4 Objetivos	16
1.5 Justificación	16
1.6 Limitaciones y Alcance	17
Capítulo II	
2. Marco Teórico	18
2.1 Sistemas Distribuidos	18
2.1.1 Definición	18
2.1.2 Características	18
2.1.3 Modelos	19
2.1.4 Cliente – Servidor	20
2.1.4.1 Categoría de Servidores	20
2.1.4.2 Componentes de Software	21
2.1.4.3 Arquitecturas Cliente / Servidor	21
2.1.4.4 Clasificación de los sistemas cliente servidor	22
2.1.5 Middleware	24
2.1.5.1 Características	25
2.1.6 Web Services	25
2.2 Integración de Sistemas Empresariales	28
2.2.1 Arquitecturas de Integración	28
2.2.1.1 Arquitectura Orientada a Servicios (SOA)	28
2.2.1.1.1 Definición	28

2.2.1.1.2	Diseño y Desarrollo de SOA	29
2.2.1.1.3	Beneficios	30
2.2.1.1.4	Diferencias con otras arquitecturas	30
2.2.1.2	Patrones de Integración (EAI)	31
2.2.1.2.1	Definición	31
2.2.1.2.2	Patrones de Diseño para servicios de mensajería	32
2.2.1.2.3	Patrones de Diseño para la manipulación de mensajes	36
2.2.1.2.4	Patrones de Diseño para servicios de orquestación	38
2.2.2	Herramientas de Integración	39
2.2.2.1	Oracle Service Bus	39
2.2.2.2	Enterprise Service Bus	40
2.2.2.2.1	¿Qué es un ESB?	40
2.2.2.2.2	Características	41
2.2.2.2.3	Beneficios	42
2.2.2.3	Apache ServiceMix	43
2.3	Industria Textil	45
2.3.1	Definición	45
2.3.2	Instituciones y redes de apoyo	46
2.3.3	Elaboración de Productos Textiles	47
2.3.4	Importancia de la producción Textil	48

Capítulo III

3.	Estado del Arte	50
3.1	Casos de Estudio	50
3.2	Selección y Justificación de la arquitectura para la solución	54
3.2.1	Arquitecturas a Evaluar	54
3.2.2	Criterios de Evaluación de la arquitectura	54
3.2.3	Análisis Comparativo de Modelos	55
3.3	Selección y Justificación de la herramienta para la solución	57
3.3.1	Herramientas a Evaluar	57
3.3.2	Criterios de Evaluación de la Herramienta	57
3.3.3	Análisis comparativo de Herramientas	58
4.	Aporte Teórico	60
4.1	Análisis de Impacto	60

4.2	Casos de Uso del Negocio	62
4.3	Casos de Uso del Sistema	78
4.4	Diagrama Funcional del Negocio	84
4.5	Arquitectura Bus	88
4.6	Modelo de Datos	89
5.	Aporte Práctico.....	90
6.	Conclusión.....	107
Bibliografía.....		108

Índice de Figuras

Figura 1.	Esquema Cliente Servidor	23
Figura 2.	Representación Distribuida	23
Figura 3.	Representación Remota	23
Figura 4.	Lógica Distribuida	24
Figura 5.	Gestión Remota de Datos	24
Figura 6.	Base de Datos Distribuidas	24
Figura 7.	Web Services	27
Figura 8.	Canal de Mensaje	32
Figura 9.	Content – Based Router	33
Figura 10.	Filtro de Mensajes	33
Figura 11.	Lista de Suscriptores	34
Figura 12.	Splitter.	34
Figura 13.	Aggregator	35
Figura 14.	Sistema de Mensajería	35
Figura 15.	Transformación de Mensajes	36
Figura 16.	Incrustación de Mensajes	36
Figura 17.	Content – Enricher	37
Figura 18.	Content – Filter	37
Figura 19.	Normalizador	38

Figura 20. ESB	40
Figura 21. Beneficios del uso de un ESB	42
Figura 22. Apache ServiceMix	43
Figura 23. Roles Entorno de Negocio	61
Figura 24. Trabajadores del Negocio	61
Figura 25. Modelo Casos de Uso	62
Figura 26. Diagrama de Actividades CUN Venta	65
Figura 27. Diagrama de Actividades CUN Almacenaje	68
Figura 28. Diagrama de Actividades CUN Producción.....	71
Figura 29. Diagrama de Actividades CUN Compra	74
Figura 30. Diagrama de Actividades CUN Distribución	77
Figura 31. Diagrama Funcional del proceso de Compras	84
Figura 32. Diagrama Funcional del proceso de Venta	85
Figura 33. Diagrama Funcional del proceso de Almacén	86
Figura 34. Diagrama Funcional del proceso de Distribución	87
Figura 35. Diagrama Arquitectura Bus	88
Figura 35. Diagrama Modelo de Datos	89

Índice de Cuadros

Cuadro 1. Rango evaluaciones Arquitecturas	55
Cuadro 2. Evaluación Arquitecturas.	55
Cuadro 3. Rango evaluación Herramientas	58
Cuadro 4. Evaluación Herramientas	58

CAPÍTULO I

1. INTRODUCCIÓN

La integración de aplicaciones y procesos de negocio es una prioridad para la gran mayoría de las empresas de hoy en día, pocas son las aplicaciones empresariales que existen solas, por lo general la mayoría se encuentran conectadas a otras aplicaciones y servicios ya sea compartiendo o referenciando los mismos datos. Partiendo de esta necesidad es que surge el concepto de “Integración de Aplicaciones Empresariales” o EAI (siglas en inglés de enterprise application integration) que se define como el uso de software y principios de arquitectura de sistemas para integrar un conjunto de aplicaciones dentro de cualquier empresa. Durante varias generaciones, los sistemas de las empresas han servido para un propósito específico a un único usuario o grupo de usuarios, los cuales actúan como la interfaz de dicho sistema con el resto de la organización, con lo cual se ha limitado su conexión con otros sistemas modernos o más amplios en la empresa. Uno de los retos que encaran las organizaciones modernas es dar a sus empleados información completa en tiempo real. Muchas de las aplicaciones en uso actualmente se apoyan en tecnologías antiguas, por lo cual esos sistemas enfrentan dificultades a la hora de mover esta información entre las aplicaciones.

EAI, como una disciplina, busca solventar muchos de esos problemas, así como crear nuevos paradigmas para ciertamente mejorar las organizaciones, tratando de trascender el objetivo de conectar las aplicaciones individuales para buscar ser un mecanismo de incrementar el conocimiento de la organización y crear ventajas competitivas futuras a la empresa.

1.1 ANTECEDENTES

El día 26 de Febrero de 1998 por Escritura Pública otorgada ante el Notario Dr. Murguía Cavero se constituyó la empresa ADALGINO TEXTIL SOCIEDAD ANÓNIMA CERRADA cuya sigla comercial es ADALTEX S.A.C. es una empresa relativamente joven ya que anteriormente funcionaba como persona Natural con Negocio desde el año 1987 a1998 produciendo telas en poliéster, algodón y deportivas en general.

Durante los primeros años hubo una muy buena organización y comunicación entre las áreas de la empresa. Los usuarios de los sistemas manejaban bien su información y era fácil poder transmitirla entre ellos. Sin embargo, de acuerdo al crecimiento y exigencia del mercado no es tan eficaz la transmisión de información. Actualmente se tienen distintas aplicaciones por módulo, sin que estas puedan comunicarse entre sí para la obtención de datos e información.

Las aplicaciones de negocios rara vez funcionan en aislamiento. Los usuarios esperan obtener acceso instantáneo a todas las funciones de negocio que una empresa puede ofrecer independientemente del sistema o grupo de sistemas donde éstas puedan residir. De ahí la necesidad de interconectar aplicaciones separadas para lograr una solución integrada que responda a los requerimientos de los usuarios.

La integración de arquitecturas tecnológicas diferentes no es una tarea fácil, muchos más cuando en la mayoría de los casos existe un número de conflictos mayor que el número de soluciones. Determinar si un tipo de arquitectura tecnológica es adecuada para una organización puede tardar meses e incluso años e implica evaluar cómo dicha arquitectura va respondiendo a los nuevos requerimientos y cambios inevitables. Desafortunadamente, no existe un "recetario" de soluciones de integración empresarial. La mayoría de los vendedores ofrecen metodologías y mejores prácticas, pero estas instrucciones tienden a ser muy orientadas hacia el conjunto de herramientas proporcionadas por el mismo vendedor.

Los patrones son una forma de capturar el conocimiento de expertos en campos donde no hay una respuesta sencilla, como la arquitectura de aplicaciones, diseño orientado a objetos, o de integración orientado a mensajes. Cada modelo plantea un

problema de diseño específico, habla de las consideraciones en torno al problema, y presenta una elegante solución que equilibre las distintas fuerzas de instrucciones.

En la mayoría de los casos, la solución no es la primera aproximación que viene a la mente. Como resultado, cada modelo incorpora la base de la experiencia que los desarrolladores de alto nivel de integración y los arquitectos han ganado durante la construcción de soluciones y aprendiendo de sus errores. Esto implica que no "inventó" los patrones, los patrones no se inventan, sino que se descubren y observan en la práctica real en algún campo.

1.2 DEFINICIÓN PROBLEMÁTICA:

Vivimos en el mundo de las telecomunicaciones dónde el principal activo de toda organización es la Información, aquel elemento base para la toma de decisiones, generador de ventajas competitivas y prerequisite fundamental para alcanzar el éxito en el mercado mundial actual. Sin embargo en la actualidad la gran mayoría de las organizaciones cuenta con una enorme cantidad de información dispersa en diferentes repositorios de datos y consumidas por sistemas y/o aplicaciones independientes, muchas de las cuales desarrolladas en tecnologías antiguas difíciles de integrar con los sistemas modernos e incluso entre ellas mismas ocasionando problemas de desaprovechamiento de la información y que a la larga genera grandes costos a las organizaciones.

Algunos de los medios de almacenamiento más comunes que podemos encontrar en las organizaciones son por ejemplo: archivos planos, documentos word, excel, aplicaciones o sistemas empresariales desarrollados en diferentes tecnologías (Java, Visual Basic, .Net, etc); diferentes bases de datos (Oracle, SQL Server, MySql, DB2, etc), entre otros. Entonces surge la interrogante: ¿Cómo integrar toda esta información a través de un único canal de comunicación que pueda ser utilizada y compartida por todas las aplicaciones que posee una organización?

En respuesta a la pregunta anterior, el mercado actual ofrece muchas alternativas y herramientas, algunas libres y otras licenciadas, sin embargo la mayoría de ellas

basadas en los mismos conceptos: Crear una arquitectura ágil, escalable y completamente distribuida por toda la organización. Algunas de éstas herramientas, las más conocidas y utilizadas, tenemos: IBM WebSphere Enterprise Service Bus (ESB), Oracle Service Bus (OSB), JBoss ESB, Apache ServiceMix, otros.

Sin embargo, lograr que todos los sistemas de una organización trabajen en forma coordinada compartiendo la misma fuente de información no es tarea fácil, debido a que existen muchos factores que la dificultan, así tenemos:

- Constante cambio (nuevos requerimientos funcionales y no funcionales).
- Problemas de conectividad.
- Problemas de gobernabilidad de TI.
- Falta de experiencia en integración de Sistemas.
- Proteccionismo de la información.
- Plataformas tecnológicas diferentes e incompatibles.
- Problemas de hardware.
- Otros.

Centrándonos en nuestro caso de estudio, Adaltex cuenta con diferentes sistemas de información distribuidos a lo largo de sus departamentos o áreas de trabajo los cuales en su momento fueron desarrollados para dar soporte a tareas específicas en alguna área de la organización. Entre éstos sistemas tenemos:

- Sistema de Contabilidad y Finanzas
- Sistema de Almacén
- Sistema de Ventas
- Sistema de Compras

Estos sistemas han venido operando sin problemas en forma independiente y aislados unos de otros durante muchos años, sin embargo con el crecimiento de la organización y de sus competidores, sus necesidades han ido cambiando, hoy en día Adaltex necesita contar con sistemas ágiles y flexibles capaces de integrarse y comunicarse entre sí (capacidades que los sistemas actuales no poseen), que provean

información real y oportuna permitiendo a la organización tomar decisiones acertadas que le generen valor respecto a sus competidores.

Durante los últimos años Adaltex viene invirtiendo considerables sumas de dinero al tratar de mejorar sus sistemas de información y tratar de hacer que éstos se lleguen a comunicar entre sí, sin embargo todo esfuerzo parece no ser suficiente debido a que siempre surgen nuevos problemas y requerimientos no previstos. Situándonos en este escenario, es que proponemos a la empresa Adaltex dar solución a su problema implementando una plataforma de comunicación integrada basada en los estándares SOA, atacando el problema de manera general y no de manera específica como lo venía haciendo sin obtener los resultados esperados.

1.3 FORMULACIÓN DEL PROBLEMA:

Problema General:

La falta de integración entre los sistemas informáticos de ADALTEX impide el óptimo procesamiento de información para la toma de decisiones estratégicas, generando así un uso ineficiente de los recursos de la organización, pérdidas económicas y desventajas competitivas frente a sus competidores.

Problemas específicos:

- Pérdida de tiempo para el traslado de información de un área a otro.
- La información que se traslada entre un área y otra puede ser no tan exacta.
- No existe un intercambio de información entre los sistemas internos de la organización.
- No existe una plataforma de comunicación flexible que permita integrar nuevas aplicaciones independientemente de la tecnología utilizada en su desarrollo.
- Bajo grado de escalabilidad de los sistemas internos de organización.
- Los componentes de las aplicaciones no son reutilizables.

1.4 OBJETIVOS

Objetivo general:

Integrar los sistemas internos de la empresa ADALTEX con una arquitectura flexible y adaptable a las tecnologías utilizadas en el desarrollo de sistemas informáticos, haciendo posible que la integración y comunicación evite los cuellos de botella y el uso ineficiente de los recursos informáticos.

Objetivos específicos:

- Lograr el paso de información entre las áreas en tiempo real.
- Asegurar que la información transmitida entre las áreas sea la correcta.
- Aprovechar eficientemente la información existente en la organización.
- Implementar una plataforma de comunicación basado en el estándar SOA.
- Mejorar el grado de escalabilidad entre los sistemas internos de una organización.
- Lograr que los componentes de las aplicaciones sean reutilizables.
- Lograr que el intercambio de información entre las áreas de negocio sea en tiempo real.

1.5 JUSTIFICACIÓN

Las infraestructuras cada vez más fragmentadas y complejas están limitando la capacidad de TI para cubrir las necesidades comerciales. Muchas organizaciones han heredado sistemas tradicionales inconexos y aplicaciones en paquete, tal es el caso de la empresa ADALTEX en donde una gran proporción de sus aplicaciones nunca fueron diseñadas para la interoperabilidad, integración ni reutilización de la información.

El resultado de todo esto es que la mayor parte del presupuesto de TI está destinado al mantenimiento de la actual infraestructura de TI y solo una pequeña porción se encuentra disponible para nuevas funcionalidades a fin de impulsar nuevas oportunidades de negocio.

Es por esta razón que realizamos una propuesta de integración de los sistemas aplicados en el negocio de la empresa ADALTEX.

1.6 ALCANCE Y LIMITACIONES

La presente tesina es de tipo Teórico y Práctico a la vez, pues se va a averiguar, aprender y mejorar nuestros conocimientos en lo que respecta a la integración de sistemas de las distintas áreas de negocio de la empresa ADALTEX para realizar sus funciones.

En el capítulo I de esta Tesina se realiza la Introducción y definición del tema elegido, indicando el problema visto y los objetivos a perseguir.

En el capítulo II se desarrolla el Marco Teórico, básico para que el lector se adentre en el tema de la Tesina.

En el capítulo III se elabora el Estado de Arte, en el cual se verán algunos modelos útiles para resolver el problema de la tesina. Se elige uno de los modelos vistos, la elección se basa en la comparación de los mismos.

En el capítulo IV se realiza el Aporte Teórico, donde se pondrá en práctica la elección del modelo o herramienta elegida en el estado del arte.

La implementación a desarrollar está basada en el caso de una empresa textilera, sin embargo, puede ser aplicado a otras empresas de rubros similares.

CAPÍTULO II

2. MARCO TEÓRICO

2.1 SISTEMAS DISTRIBUIDOS

2.1.1 Definición:

Sistemas cuyos componentes hardware y software, que están en ordenadores conectados en red, se comunican y coordinan sus acciones mediante el paso de mensajes, para el logro de un objetivo. Se establece la comunicación mediante un protocolo prefijado por un esquema cliente-servidor.

2.1.2 Características:

- **Concurrencia.-** Esta característica de los sistemas distribuidos permite que los recursos disponibles en la red puedan ser utilizados simultáneamente por los usuarios y/o agentes que interactúan en la red.
- **Carencia de reloj global.-** Las coordinaciones para la transferencia de mensajes entre los diferentes componentes para la realización de una tarea, no tienen una temporización general, esta más bien distribuida a los componentes.
- **Fallos independientes de los componentes.-** Cada componente del sistema puede fallar independientemente, con lo cual los demás pueden continuar ejecutando sus acciones. Esto permite el logro de las tareas con mayor efectividad, pues el sistema en su conjunto continua trabajando.

2.1.3 Modelos:

- **Procesamiento central (Host).**- Uno de los primeros modelos de ordenadores interconectados, llamados centralizados, donde todo el procesamiento de la organización se llevaba a cabo en una sola computadora, normalmente un Mainframe, y los usuarios empleaban sencillos ordenadores personales.

Los problemas de este modelo son:

Cuando la carga de procesamiento aumentaba se tenía que cambiar el hardware del Mainframe, lo cual es más costoso que añadir más computadores personales clientes o servidores que aumenten las capacidades.

- **Grupo de Servidores.**- Otro modelo que entró a competir con el anterior, también un tanto centralizado, son un grupo de ordenadores actuando como servidores, normalmente de archivos o de impresión, poco inteligentes para un número de Minicomputadores que hacen el procesamiento conectados a una red de área local.

Los problemas de este modelo son:

Podría generarse una saturación de los medios de comunicación entre los servidores poco inteligentes y los minicomputadores, por ejemplo cuando se solicitan archivos grandes por varios clientes a la vez, podían disminuir en gran medida la velocidad de transmisión de información.

- **La Computación Cliente Servidor.**- Este modelo, que predomina en la actualidad, permite descentralizar el procesamiento y recursos, sobre todo, de cada uno de los servicios y de la visualización de la Interfaz Gráfica de Usuario. Esto hace que ciertos servidores estén dedicados solo a una aplicación determinada y por lo tanto ejecutarla en forma eficiente.

2.1.4 Cliente-Servidor

Sistema donde el cliente es una máquina que solicita un determinado servicio y se denomina servidor a la máquina que lo proporciona. Los servicios pueden ser:

- Ejecución de un determinado programa.
- Acceso a un determinado banco de información.
- Acceso a un dispositivo de hardware.

Es un elemento primordial, la presencia de un medio físico de comunicación entre las máquinas, y dependerá de la naturaleza de este medio la viabilidad del sistema.

2.1.4.1 Categorías de Servidores:

A continuación se presenta una lista de los servidores más comunes:

- **Servidores de archivos.-** Proporciona archivos para clientes. Si los archivos no fueran tan grandes y los usuarios que comparten esos archivos no fueran muchos, esto sería una gran opción de almacenamiento y procesamiento de archivos. El cliente solicita los archivos y el servidor los ubica y se los envía.
- **Servidores de Base de Datos.-** Son los que almacenan gran cantidad de datos estructurados, se diferencian de los de archivos pues la información que se envía está ya resumida en la base de datos. Ejemplo: El Cliente hace una consulta, el servidor recibe esa consulta (SQL) y extrae solo la información pertinente y envía esa respuesta al cliente.
- **Servidores de Software de Grupo.-** El software de grupo es aquel, que permite organizar el trabajo de un grupo. El servidor gestiona los datos que dan soporte a estas tareas. Por ejemplo: almacenar las listas de correo electrónico. El Cliente puede indicarle, que se ha terminado una tarea y el servidor se lo envía al resto del grupo.
- **Servidores WEB.-** Son los que guardan y proporcionan Páginas HTML. El cliente desde un browser o link hace un llamado de la página y el servidor recibe el mensaje y envía la página correspondiente.

- **Servidores de correo.-** Gestiona el envío y recepción de correo de un grupo de usuarios (el servidor no necesita ser muy potente). El servidor solo debe utilizar un protocolo de correo.
- **Servidor de objetos.-** Permite almacenar objetos que pueden ser activados a distancia. Los clientes pueden ser capaces de activar los objetos que se encuentran en el servidor.
- **Servidores de impresión.-** Gestionan las solicitudes de impresión de los clientes. El cliente envía la solicitud de impresión, el servidor recibe la solicitud y la ubica en la cola de impresión, ordena a la impresora que lleve a cabo las operaciones y luego avisa a la computadora cliente que ya acabo su respectiva impresión.
- **Servidores de aplicación.-** Se dedica a una única aplicación. Es básicamente una aplicación a la que pueden acceder los clientes.

2.1.4.2 Componentes de Software:

Se distinguen tres componentes básicos de software:

- **Presentación.-** Tiene que ver con la presentación al usuario de un conjunto de objetos visuales y llevar a cabo el procesamiento de los datos producidos por el mismo y los devueltos por el servidor.
- **Lógica de aplicación.-** Esta capa es la responsable del procesamiento de la información que tiene lugar en la aplicación.
- **Base de datos.-** Esta compuesta de los archivos que contienen los datos de la aplicación.

2.1.4.3 Arquitecturas Cliente / Servidor

A continuación mostramos las arquitecturas cliente-servidor más populares:

- **Arquitectura Cliente-Servidor de Dos Capas.**

Consiste en una capa de presentación y lógica de la aplicación; y la otra de la base de datos. Normalmente esta arquitectura se utiliza en las siguientes situaciones:

- Cuando se requiera poco procesamiento de datos en la organización.
- Cuando se tiene una base de datos centralizada en un solo servidor.
- Cuando la base de datos es relativamente estática.
- Cuando se requiere un mantenimiento mínimo.

- **Arquitectura Cliente-Servidor de Tres Capas**

Consiste en una capa de la Presentación, otra capa de la lógica de la aplicación y otra capa de la base de datos. Normalmente esta arquitectura se utiliza en las siguientes situaciones:

- Cuando se requiera mucho procesamiento de datos en la aplicación.
- En aplicaciones donde la funcionalidad este en constante cambio.
- Cuando los procesos no están relativamente muy relacionados con los datos.
- Cuando se requiera aislar la tecnología de la base de datos para que sea fácil de cambiar.
- Cuando se requiera separar el código del cliente para que se facilite el mantenimiento.
- Esta muy adecuada para utilizarla con la tecnología orientada a objetos.

2.1.4.4 Clasificación de los sistemas cliente servidor:

A continuación mostramos la clasificación de de los sistemas cliente/servidor de acuerdo al nivel de abstracción del servicio que ofrecen:

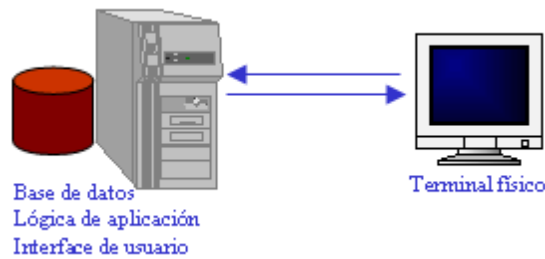


Figura 1. Esquema Cliente Servidor

Representación distribuida.- La interacción con el usuario se realiza en el servidor, el cliente hace de pasarela entre el usuario y el servidor.

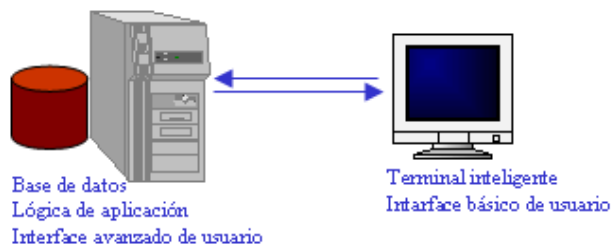


Figura 2. Representación Distribuida

Representación Remota.-La lógica de la aplicación y la base de datos se encuentran en el servidor. El cliente recibe y formatea los datos para interactuar con el usuario.

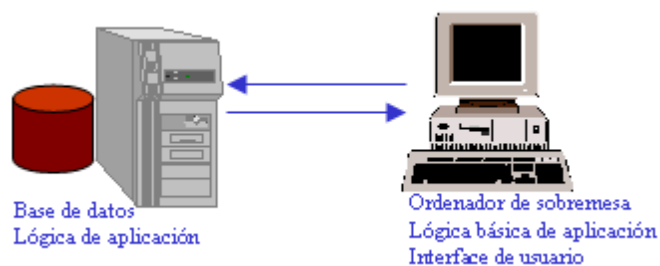


Figura 3. Representación Remota

Lógica Distribuida.- El cliente se encarga de la interacción con el usuario y de algunas funciones triviales de la aplicación. Por ejemplo controles de rango de campos, campos obligatorios, etc. Mientras que el resto de la aplicación, junto con la base de datos, están en el servidor.

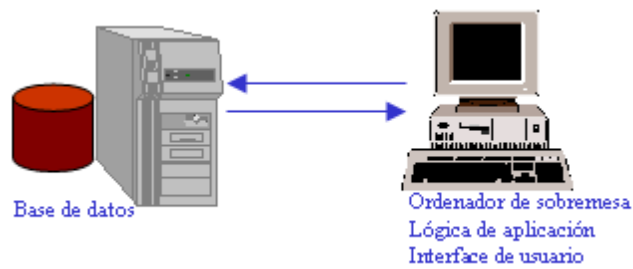


Figura 4. Lógica Distribuida

Gestión Remota de Datos.- El cliente realiza la interacción con el usuario y ejecuta la aplicación y el servidor es quien maneja los datos.

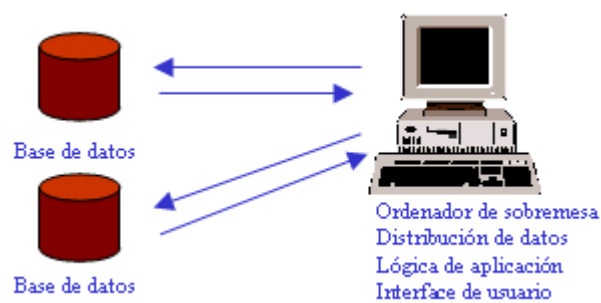


Figura 5. Gestión remota de Datos

Base de Datos Distribuidas.- El cliente realiza la interacción con el usuario, ejecuta la aplicación, debe conocer la topología de la red, así como la disposición y ubicación de los datos. Se delega parte de la gestión de la base de datos al cliente.

Cliente servidor a tres niveles.- El cliente se encarga de la interacción con el usuario, el servidor de la lógica de aplicación y la base de datos puede estar en otro servidor.

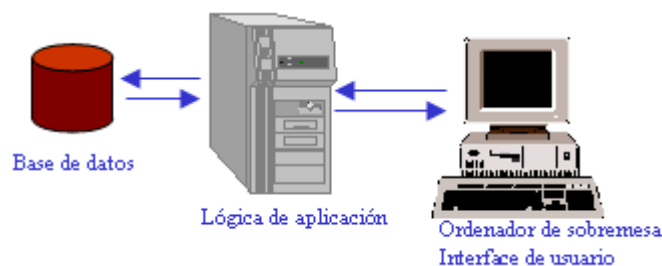


Figura 6. Base de Datos Distribuidas

2.1.5 MIDDLEWARE

Capa de software intermedio entre el cliente y el servidor. Es la capa de software que nos permiten gestionar los mecanismos de comunicaciones. Ejemplo si se hace la petición de una página web desde un browser en el cliente, el middleware determina la ubicación y envía una petición para dicha página. El servidor Web, interpreta la petición y envía la página al software intermedio, quien la dirige al navegador de la máquina cliente que la solicitó.

Existen dos tipos:

- **Software intermedio general.** Servicios generales que requieren todos los clientes y servidores, por ejemplo: software para las comunicaciones usando el TCP/IP, software parte del sistema operativo que, por ejemplo, almacena los archivos distribuidos, software de autenticación, el software intermedio de mensajes de clientes a servidores y viceversa.
- **Software intermedio de servicios.** Software asociado a un servicio en particular, por ejemplo: software que permite a dos BD conectarse a una red cliente/servidor (ODBC: Conectividad abierta de BD), software de objetos distribuidos, por ejemplo la tecnología CORBA permite que objetos distribuidos creados en distintos lenguajes coexistan en una misma red (intercambien mensajes), software intermedio para software de grupo, software intermedio asociado a productos de seguridad específicas (Conexiones Seguras: Sockets), etc.

2.1.5.1 Características:

Independiza el servicio de su implantación, del sistema operativo y de los protocolos de comunicaciones.

- Permite la convivencia de distintos servicios en un mismo sistema.
- Permite la transparencia en el sistema.
- Modelo tradicional: Monitor de teleproceso o CICS, Tuxedo, Encina.
- Modelo OO: CORBA.

2.1.6 WEB SERVICES

Cuando hablamos de Web services (Servicios Web), hablamos de una tecnología muy reciente, y por su utilidad también ha sido una de las más difundidas. Se puede decir que los Web services es una de las tendencias clave en la integración de sistemas, como también es de gran ayuda en las nuevas metodologías para compartir datos de distintas fuentes por sus características de control, seguridad y sus métodos estándar actuales. Para ello, ha debido pasar por varias etapas de crecimiento, lo que ha dado paso a estudios con diversos planteamientos y postulados, lo que colocan a los Web services como uno de los estándares más importantes junto a XML en los próximos años.

Los Web services, son aplicaciones que intercambian datos entre sí con el objetivo de ofrecer servicios, en el cual existen proveedores que prestan sus servicios como procedimientos remotos y usuarios clientes que solicitan un servicio específico llamando a estos procedimientos a través de la Web, los que son comúnmente de tamaño pequeño y formados por varios componentes que comparten su información de forma segura y estructurada con las demás aplicaciones, en forma rápida y estandarizada. Estas aplicaciones o servicios son publicados en directorios, los que pueden ser invocados y ejecutados vía HTTP, en el cual su transporte es por el lenguaje estándar XML.

Otra definición es dada a los Servicios WEB, es que estos son conjunto de aplicaciones o de tecnologías con capacidad para interoperar en la Web. Estas aplicaciones o tecnologías intercambian datos entre sí con el objetivo de ofrecer unos servicios. Los proveedores ofrecen sus servicios como procedimientos remotos y los usuarios solicitan un servicio llamando a estos procedimientos a través de la Web. La utilización de los Web services ha sido potenciada por tratarse de un estándar y que a diferencia de otras tecnologías de integración, permite compartir funcionalidades entre sistemas heterogéneos de forma transparente, mediante el intercambio de datos vía XML, siendo su único requisito el establecimiento de conexiones TCP/IP. Los Web services proporcionan mecanismos de comunicación estándares entre diferentes aplicaciones, que interactúan entre sí para presentar información dinámica al usuario. Para proporcionar interoperabilidad y extensibilidad entre estas aplicaciones, y que al

mismo tiempo sea posible su combinación para realizar operaciones complejas, es necesaria una arquitectura de referencia estándar. Por lo que simplifica mucho la integración de distintas aplicaciones.

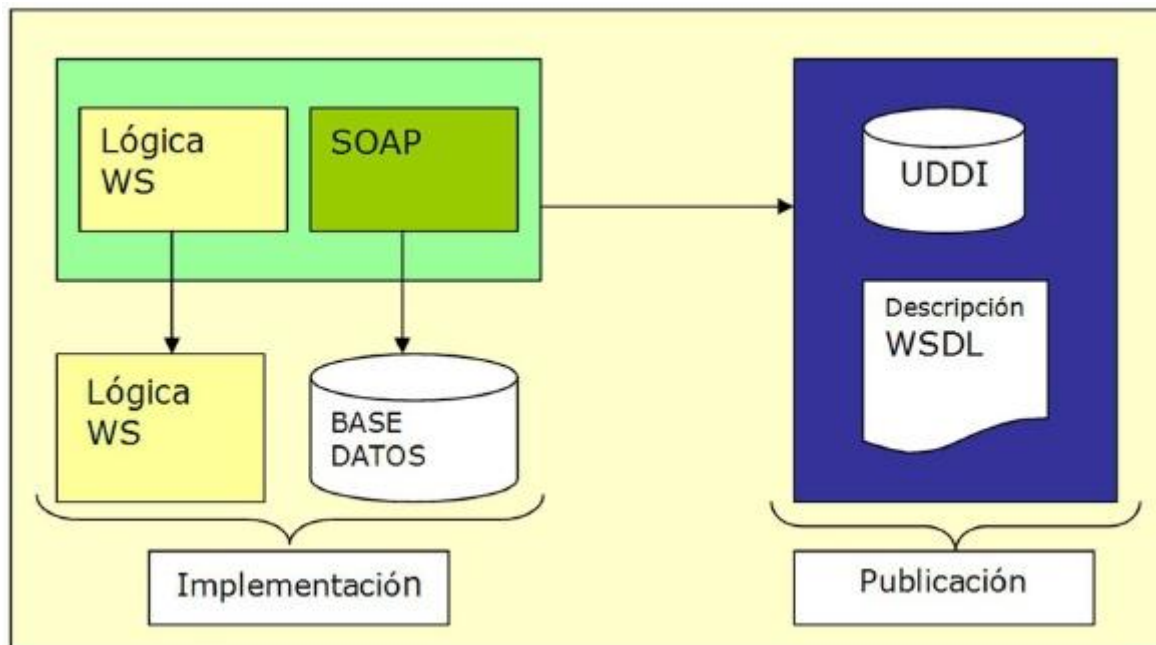


Figura 7. Web Services

2.2 INTEGRACIÓN DE SISTEMAS EMPRESARIALES

La arquitectura de las tecnologías de la información es el componente esencial sobre el que se construyen relaciones provechosas. Al igual que se necesita una amplia variedad de personas y habilidades para poder responder a las necesidades iniciales y prestar soporte continuo a un cliente dado, también las tecnologías deben integrar las posibilidades y los datos presentes en cualquier número de sistemas individuales para dar soporte a dichos clientes. Cuando esto ocurre, los resultados obtenidos pueden alterar sustancialmente la forma en que se entienden los beneficios empresariales.

Actualmente en los procesos estratégicos en las organizaciones participan múltiples aplicaciones, por lo que es de vital importancia que dichas aplicaciones estén conectadas e intercambien información entre sí. Este tipo de aplicaciones procesan los datos contenidos en los sistemas de la empresa, preparan la información y la convierten en conocimiento valioso y estratégico para la organización, agilizando la toma de decisiones.

2.2.1 ARQUITECTURAS DE INTEGRACIÓN

2.2.1.1 ARQUITECTURA ORIENTADA A SERVICIOS (SOA)

2.2.1.1.1 Definición

La arquitectura orientada a servicios de cliente (en inglés Service Oriented Architecture), es un concepto de arquitectura de software que define la utilización de servicios para dar soporte a los requisitos del negocio.

Permite la creación de sistemas de información altamente escalables que reflejan el negocio de la organización, a su vez brinda una forma bien definida de exposición e invocación de servicios (comúnmente pero no exclusivamente servicios web), lo cual facilita la interacción entre diferentes sistemas propios o de terceros.

SOA define las siguientes capas de software:

- Aplicaciones básicas - Sistemas desarrollados bajo cualquier arquitectura o tecnología, geográficamente dispersos y bajo cualquier figura de propiedad;
- De exposición de funcionalidades - Donde las funcionalidades de la capa applicativa son expuestas en forma de servicios (generalmente como servicios web);
- De integración de servicios - Facilitan el intercambio de datos entre elementos de la capa applicativa orientada a procesos empresariales internos o en colaboración;
- De composición de procesos - Que define el proceso en términos del negocio y sus necesidades, y que varía en función del negocio;
- De entrega - donde los servicios son desplegados a los usuarios finales.

SOA proporciona una metodología y un marco de trabajo para documentar las capacidades de negocio y puede dar soporte a las actividades de integración y consolidación.

2.2.1.1.2 Diseño y desarrollo de SOA

La metodología de modelado y diseño para aplicaciones SOA se conoce como análisis y diseño orientado a servicios. La arquitectura orientada a servicios es tanto un marco de trabajo para el desarrollo de software como un marco de trabajo de implementación. Para que un proyecto SOA tenga éxito los desarrolladores de software deben orientarse ellos mismos a esta mentalidad de crear servicios comunes que son orquestados por clientes o middleware para implementar los procesos de negocio. El desarrollo de sistemas usando SOA requiere un compromiso con este modelo en términos de planificación, herramientas e infraestructura.

Cuando la mayoría de la gente habla de una arquitectura orientada a servicios están hablando de un juego de servicios residentes en Internet o en una intranet, usando servicios web. Existen diversos estándares relacionados a los servicios web. Incluyen los siguientes:

- XML
- HTTP
- SOAP

- REST
- WSDL
- UDDI

Hay que considerar, sin embargo, que un sistema SOA no necesariamente necesita utilizar estos estándares para ser "Orientado a Servicios" pero es altamente recomendable su uso.

En un ambiente SOA, los nodos de la red hacen disponibles sus recursos a otros participantes en la red como servicios independientes a los que tienen acceso de un modo estandarizado. La mayoría de las definiciones de SOA identifican la utilización de Servicios Web (empleando SOAP y WSDL) en su implementación, no obstante se puede implementar SOA utilizando cualquier tecnología basada en servicios.

2.2.1.1.3 Beneficios

Los beneficios que puede obtener una organización que adopte SOA son:

- Mejora en los tiempos de realización de cambios en procesos.
- Facilidad para evolucionar a modelos de negocios basados en tercerización.
- Facilidad para abordar modelos de negocios basados en colaboración con otros entes (socios, proveedores).
- Poder para reemplazar elementos de la capa applicativa SOA sin disrupción en el proceso de negocio.
- Facilidad para la integración de tecnologías disímiles.

2.2.1.1.4 Diferencias con otras arquitecturas

Al contrario de las arquitecturas orientado a objetos, las SOAs están formadas por servicios de aplicación débilmente acoplados y altamente interoperables. Para comunicarse entre sí, estos servicios se basan en una definición formal independiente de la plataforma subyacente y del lenguaje de programación. La definición de la interfaz encapsula (oculta) las particularidades de una implementación, lo que la hace independiente del fabricante, del lenguaje de programación o de la tecnología de desarrollo (como Plataforma Java o Microsoft .NET). Con esta arquitectura, se pretende

que los componentes de software desarrollados sean muy reutilizables, ya que la interfaz se define siguiendo un estándar; así, un servicio C# podría ser usado por una aplicación Java. En este sentido, ciertos autores definen SOA como una Súper-Abstracción.

2.2.1.2 PATRONES DE INTEGRACIÓN (EAI).

2.2.1.2.1 Definición

Los patrones de integración (o EAI) definen diseños comunes (patrones) en el desarrollo de funcionalidades relacionadas con la integración de aplicaciones. Especifican una manera estándar de realizar ciertas tareas y ayudan a conocer con un lenguaje común determinadas cosas que desarrollamos habitualmente.

Los sistemas de EAI principalmente implementan dos patrones:

- **Mediación:** Donde los sistemas de EAI actúan como el canal que vincula varias aplicaciones o sistemas utilizando técnicas de encaminamiento. Cuando en una aplicación (interlocutor origen) ocurre un evento que puede ser interesante a alguna otra aplicación (interlocutor destino) esta debe notificar a un módulo de integración (servicio) del sistema EAI. Entonces el sistema de EAI propaga esos cambios a las otras aplicaciones relevantes.
- **Federación:** En este caso, el sistema EAI actúa como un consolidador de información entre varias aplicaciones. Todos los accesos al o del exterior de la organización a cualquiera de las aplicaciones son gestionados por el sistema EAI, y éste está configurado para exponer sólo la información relevante conectándose a las aplicaciones del mundo exterior, y efectuar todas las interacciones con las aplicaciones internas sin intervención del agente externo.

Ambos patrones son usados en conjunto frecuentemente. El mismo sistema EAI puede tener varias aplicaciones en mediación, mientras sirve requerimientos de agentes externos contra esas aplicaciones (federación).

2.2.1.2.2 Patrones de diseño para servicios de mensajería

- **Message Channel**

- Conectar dos aplicaciones mediante un sistema de mensajería

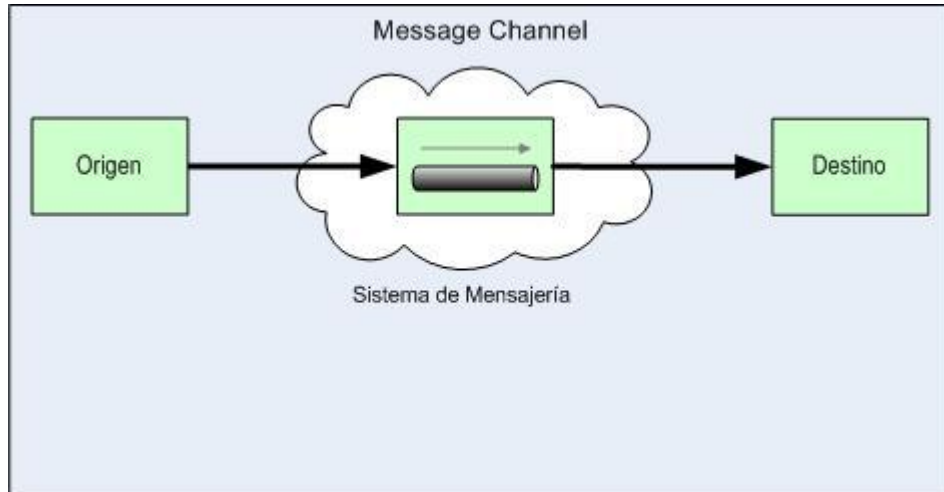


Figura 8. Canal de Mensaje

- **Pipe & Filters**

- Realizar procesos de una determinada complejidad en mensajes manteniendo independencia y flexibilidad
- Modularización y ejecución de tareas de procesamiento de mensajes en etapas Implementación (BTS)

Content-Based Router

- Examina el contenido de un mensaje para distribuirlo por diferentes canales en función de los datos que componen dicho mensaje

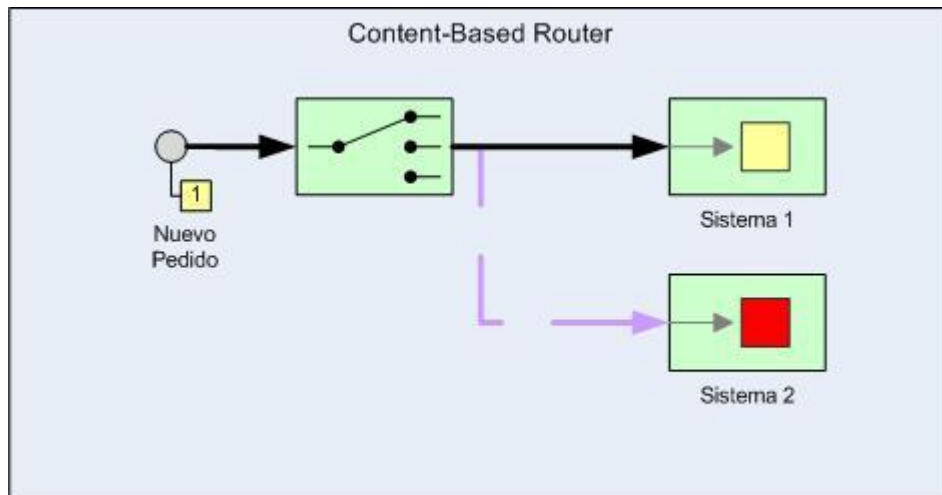


Figura 9. Content – Based Router

- **Message Filter**

- Evitar recepción de mensajes no deseados
- Crear un conjunto de criterios (filtros) de manera que si el mensaje contiene propiedades que hacen coincidir con los filtros especificados, este será enviado por un canal de salida, en caso contrario será descartado.

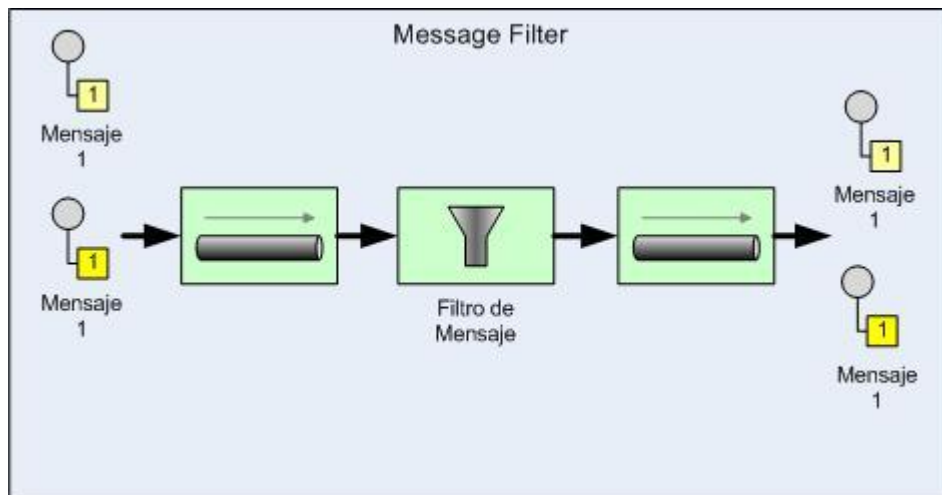


Figura 10. Filtro de Mensajes

- **Recipient List**

- Encaminar un mensaje a una lista de suscriptores del mismo.

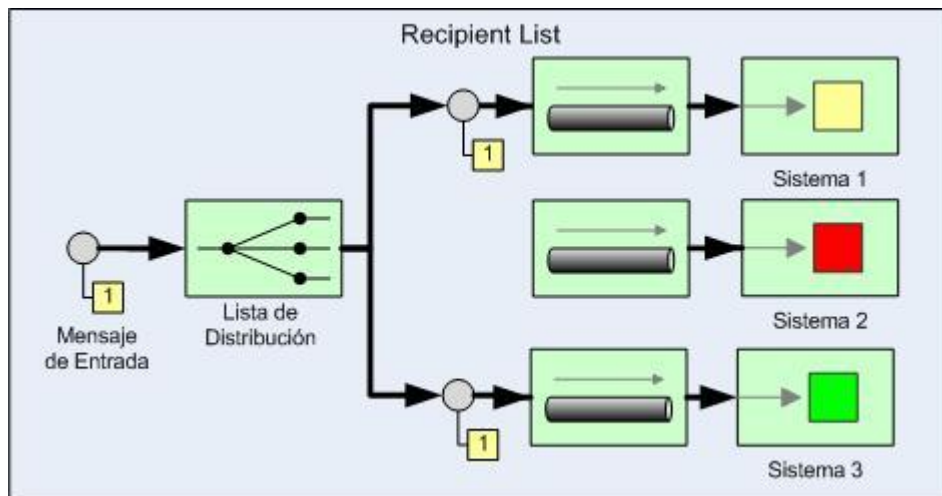


Figura 11. Lista de Suscriptores

- **Splitter**

- Descompone un mensaje que contiene una lista de elementos repetitivos, para que estos puedan ser procesados de forma individual.

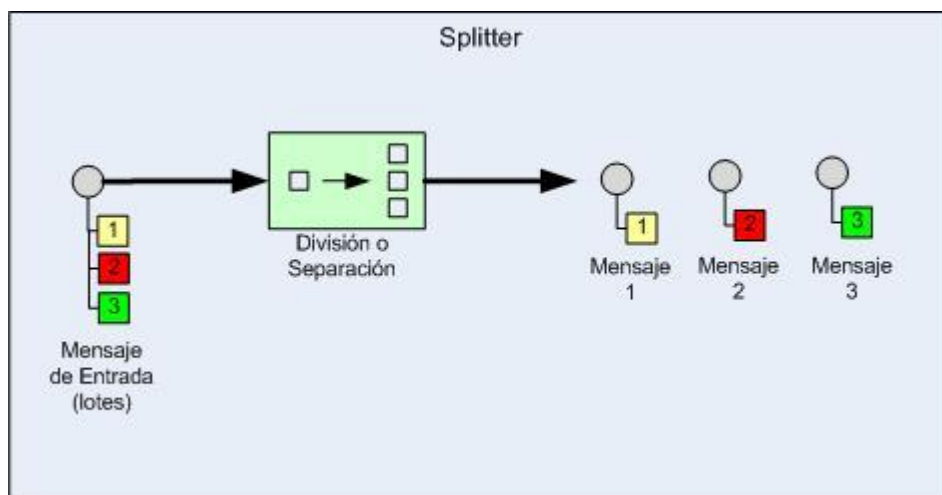


Figura 12. Splitter

Aggregator

- Combina varios mensajes individuales pero que se relacionan entre ellos en un único mensaje para que puedan ser procesados de una forma global

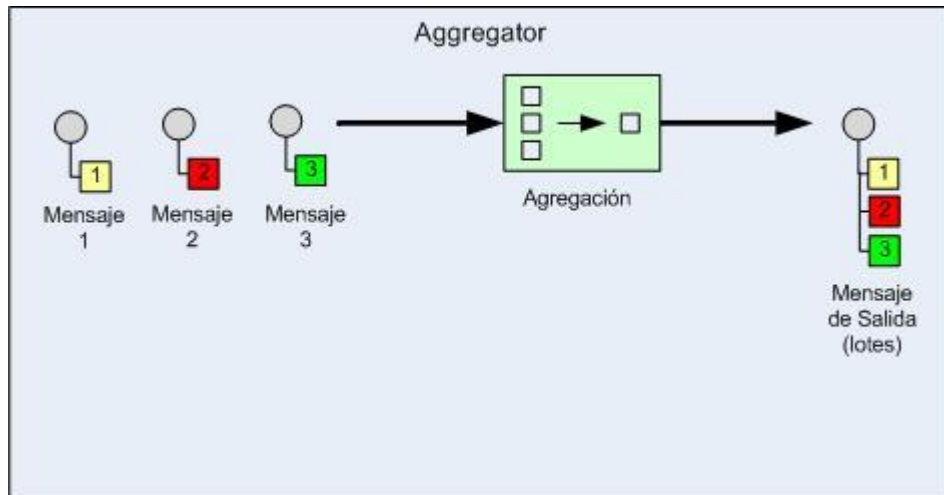


Figura 13. Aggregator

Scatter-Gather

- Conectar dos aplicaciones mediante un sistema de mensajería

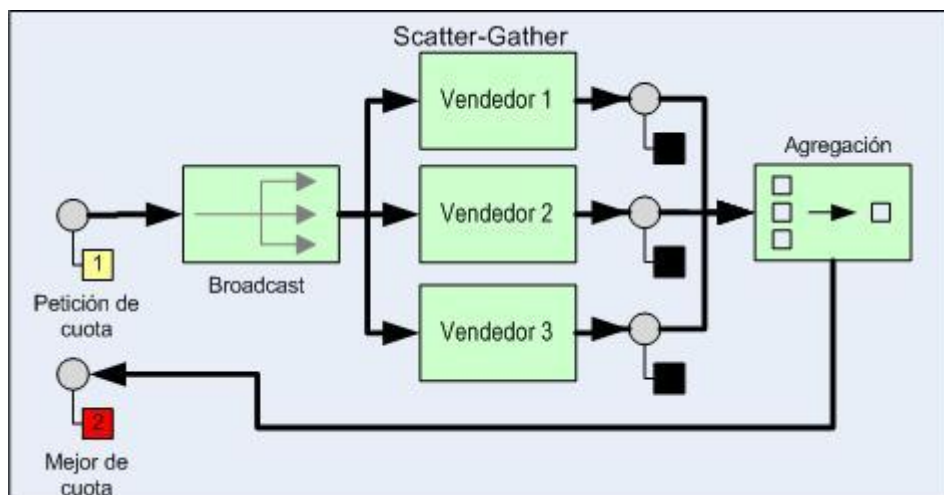


Figura 14. Sistema de Mensajería

2.2.1.2.3 Patrones de diseño para la manipulación de mensajes

- **Message Translator**

- Transformación de un mensaje en otro para que pueda ser usado en un contexto diferente (Adapter GoF)

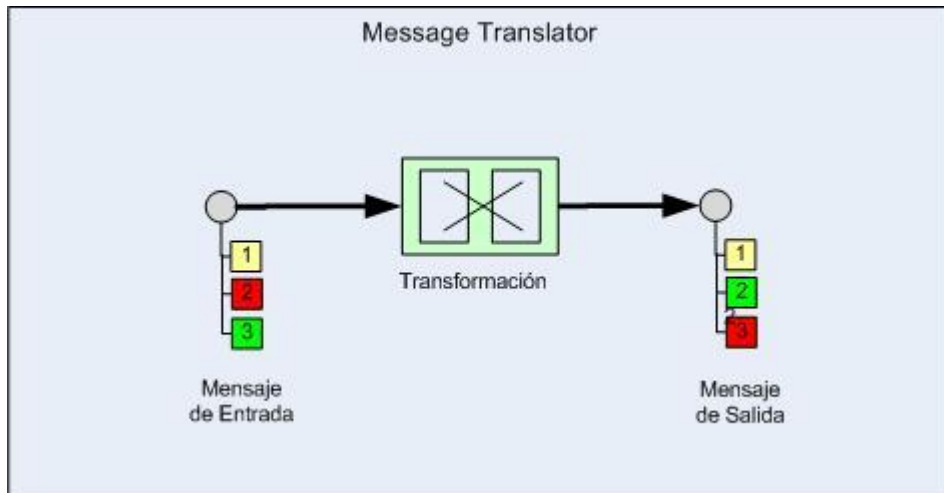


Figura 15. Transformación de Mensajes

Envelop Wrapper

- Incrustar un mensaje (funcional) como cuerpo de un mensaje (protocolo)
- Pretende no mezclar procesos funcionales con procesos tecnológicos

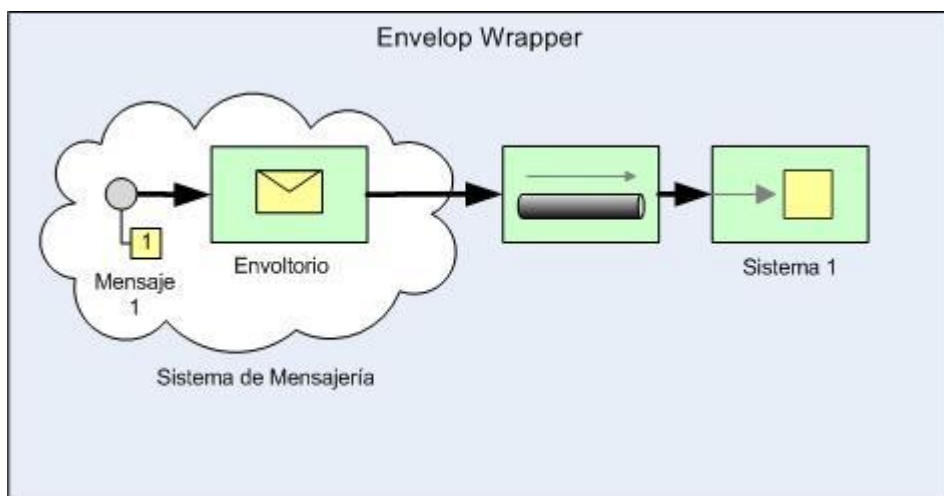


Figura 16. Incrustación de Mensajes

- **Content-Enricher**

- Añade datos necesarios en la construcción de un mensaje cuando estos no están presentes en el mensaje original

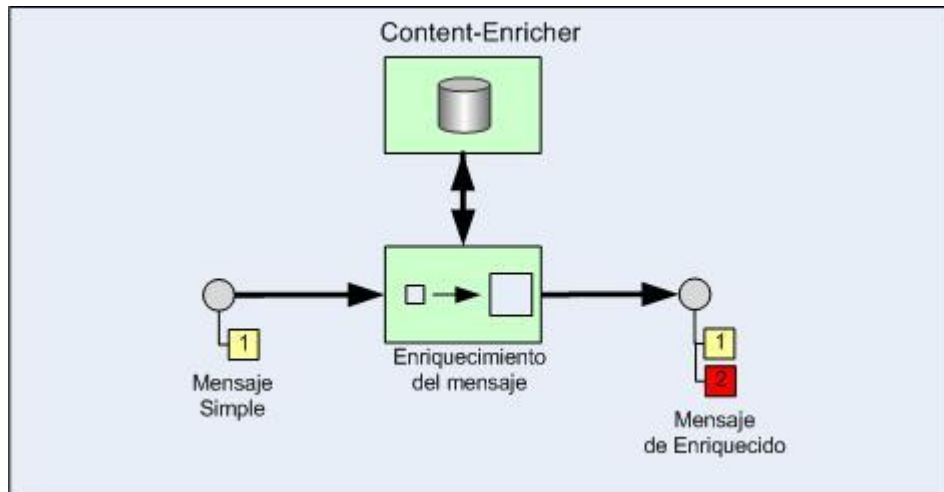


Figura 17. Content - Enricher

- **Content-Filter**

- Simplifica la estructura de un mensaje previo a su entrega cuando esta no es relevante para la aplicación destino.

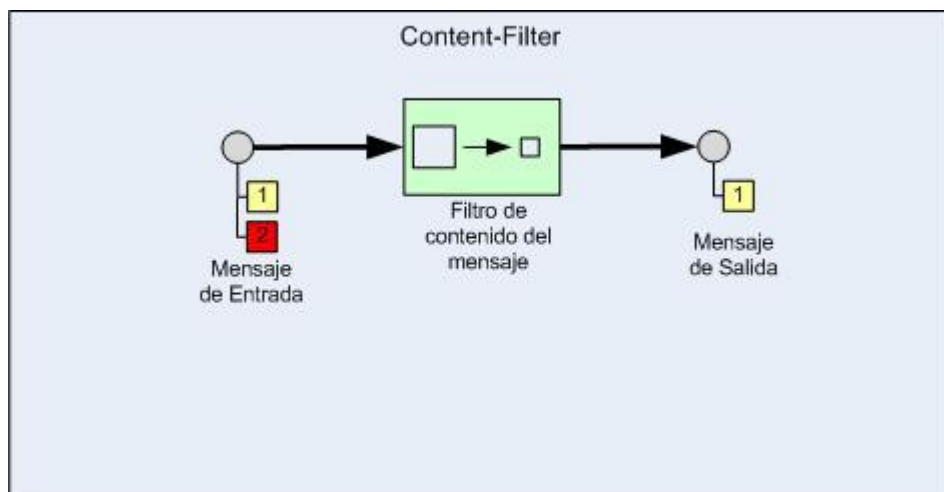


Figura 18. Content - Filter

- **Normalizer**
 - Resuelve la problemática de procesamiento de mensajes que contienen la misma semántica, pero distinto formato

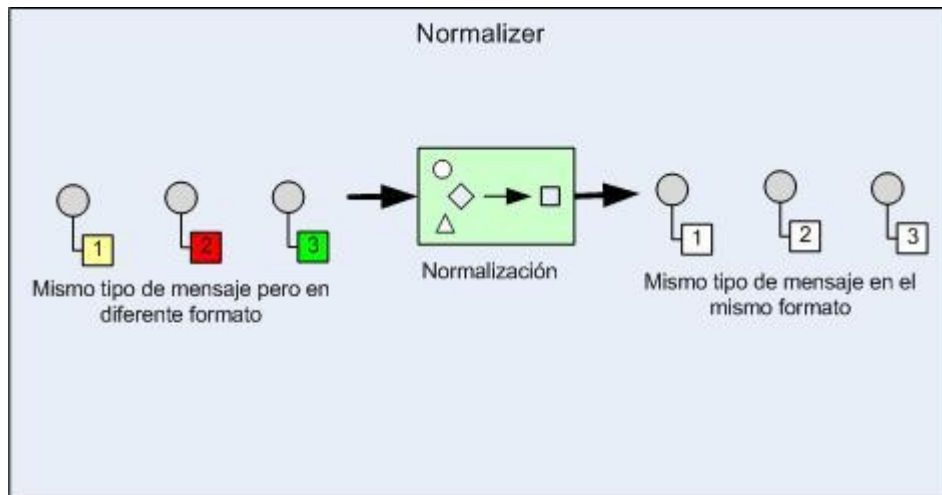


Figura 19. Normalizador

2.2.1.2.4 Patrones de diseño para servicios de orquestación

- **Convoys**
 - Este patrón se usa para asegurarse que una misma instancia de una orquestación procesa mensajes consecutivos conectados a una misma operación
- **Error Handling**
 - Añadir piezas de código por cada forma (o scope) dentro de un flujo de trabajo para gestionar errores inesperados y facilitar la recuperación de los mismos
 - Control de errores
 - Manejadores de excepciones
 - Lógica de compensación
 - Suspensión / Terminación del proceso

2.2.2 HERRAMIENTAS DE INTEGRACIÓN

2.2.2.1. ORACLE SERVICE BUS

Oracle Service Bus está diseñado para conectarse, mediar y administrar las interacciones entre

servicios heterogéneos, aplicaciones tradicionales, aplicaciones empaquetadas y múltiples instancias Enterprise Service Bus (ESB) a través de una red de servicios para toda la empresa.

Oracle Service Bus permite la integración de servicios controlados por la configuración, con ruteo basado en identidades y contenido inteligente. Mejora la productividad del desarrollador debido a la integración de servicios de código libre. Oracle Service Bus también brinda transporte nativo para aplicaciones empaquetadas y planificación de recursos empresariales líderes, junto con la conectividad a aplicaciones basadas en el servidor.

Oracle Service Bus brinda capacidades incorporadas para la virtualización de servicios, Web Service Security (WS-Security) y el cumplimiento de políticas en torno a la regulación y la agrupación de servicios a fin de cumplir con los requerimientos de Confiabilidad, Disponibilidad y Desempeño (RASP) y evitar la sobrecarga de servicios de back-end. Oracle Service Bus está creado desde cero con soporte integral para SOA, Java 2 Platform, Enterprise Edition (J2EE), y estándares.

También conocido como OSB, es un Enterprise Service Bus aplicación por parte de Oracle. Oracle Service Bus transforma arquitecturas complejas y frágiles en las redes de integración mediante la conexión ágil, la mediación y la gestión de las interacciones entre los servicios y aplicaciones. Oracle Service Bus ofrece un bajo costo, integración basada en estándares para entornos SOA de misión crítica en un rendimiento extremo y escalabilidad son los requisitos.

Características de Oracle Service Bus

- ESB de múltiples protocolos
- Integración impulsada por la configuración

- Ruteo basado en identidad y contenido
- Seguridad orientada a políticas
- Dominios ESB distribuidos
- Calidad de Servicio superior (RASP)

2.2.2.2 ENTERPRISE SERVICE BUS



Figura 20. ESB

2.2.2.2.1 ¿Qué es un ESB?

Es el proceso de conectar las aplicaciones unas con otras para intercambiar información operativa o financiera. Cuando dichos sistemas no pueden compartir su información efectivamente, se crean cuellos de botella que requieren de la intervención humana en la forma de toma de decisiones o en el ingreso mismo de la información. Con una arquitectura EAI correctamente implementada, las organizaciones pueden enfocar la mayoría de sus esfuerzos en la creación de competencias que generen valor, en lugar de enfocarse en la coordinación de labores operativas.

Durante varias generaciones, los sistemas de las empresas han servido para un propósito específico a un único usuario o grupo de usuarios, los cuales actúan como la interfaz de dicho sistema con el resto de la organización, limitando su conexión con otros sistemas modernos o más amplios en la empresa y más aún, por la creciente demanda de las empresas por compartir datos y usarlos en sus procesos sin tener que realizar cambios en sus aplicaciones o en sus estructuras de datos.

Uno de los retos que encaran las organizaciones modernas es darle a sus empleados información completa en tiempo real. Muchas de las aplicaciones en uso actualmente se apoyan en tecnologías antiguas, por lo cual esos sistemas enfrentan dificultades a la hora de mover esta información entre las aplicaciones.

EAI, como una disciplina, busca solventar muchos de esos problemas, así como crear nuevos paradigmas para, ciertamente, mejorar a las organizaciones tratando de trascender en el objetivo de conectar las aplicaciones individuales, para ser un mecanismo que incremente el conocimiento dentro de la organización y crear ventajas competitivas futuras a la empresa.

Un bus de integración (ESB) ocupa la capa de abstracción intermedia (middleware) entre los distintos sistemas de una o varias organizaciones, proporcionando mecanismos de comunicación y transformación a través de mensajería basada en estándares.

En definitiva, un ESB debe ser capaz de reemplazar todo el contacto directo entre aplicaciones, consiguiendo que todas ellas se comuniquen a través del bus.

Los ESB transmiten y reciben mensajes basados en estándares, pero deben ser capaces de transformar mensajes a formatos que sean reconocidos por las distintas aplicaciones en el caso de que sea necesario, lo que se realiza a través de adaptadores.

Además, el intercambio de mensajes debe ser independiente de la plataforma. Esto permite al ESB integrar aplicaciones que se ejecutan en diversos sistemas operativos o mainframes.

2.2.2.2 Características de un ESB

- Independientes respecto a sistemas operativos y lenguajes de programación.
- Uso de XML como lenguaje estándar de comunicación.
- Soporte de estándares de Servicios Web.
- Adaptadores para realizar la integración con aplicaciones.
- Modelo de seguridad estándar para autorizar, autenticar y auditar el uso del ESB.
- Transformación de mensajes.

- Validación de mensajes.
- Enrutamiento de mensajes aplicando reglas de negocio y en función del contenido del mensaje.
- Manipulación de excepciones.
- Soporte a encolado y mantenimiento de mensajes, si las aplicaciones no están disponibles.
- Monitorización del sistema y de la actividad de negocio (BAM).

2.2.2.2.3 Beneficios del uso de un ESB

- Integraciones más rápidas y económicas de los sistemas.
- Mayor flexibilidad ante cambios.
- Basado en estándares.
- Servicios predefinidos que ahorran costes de desarrollo.
- Más configuración que código a la hora de integrar.
- Mayor control del sistema.
- Reglas de negocio manejadas por analistas de negocio.
- Seguridad en la integración entre aplicaciones.

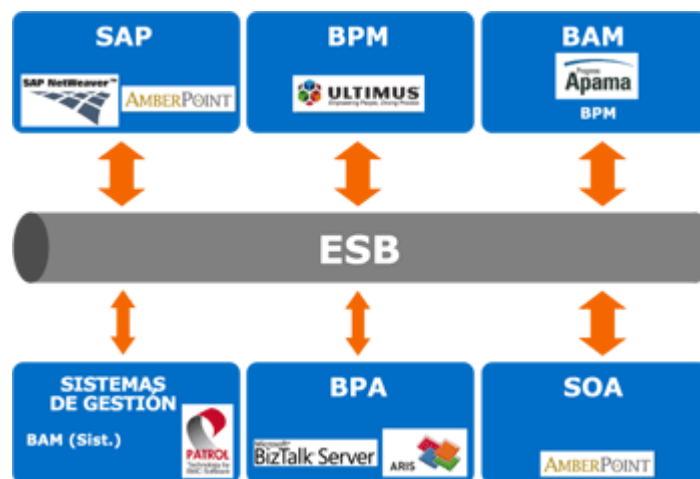


Figura 21. Beneficios del uso de un ESB

2.2.2.3 APACHE SERVICEMIX

Apache ServiceMix es una solución flexible y de código abierto contenedor de integración que unifica las características y funcionalidades de Apache ActiveMQ , Camel , CXF , ODE , Karaf en una plataforma de ejecución de gran alcance que puede utilizar para crear sus soluciones propias integraciones. Proporciona una ESB completa, lista para la empresa exclusivamente impulsado por OSGi.

Las características principales son:

- mensajería de confianza con el Apache ActiveMQ
- mensajería, enrutamiento y patrones de integración de la empresa con Apache Camel
- la integración de acoplamiento flexible entre todos los otros componentes con Apache ServiceMix RMN incluyendo eventos rico, de mensajería y de la API de Auditoría
- OSGi servidor basado en tiempo de ejecución impulsado por Apache Karaf

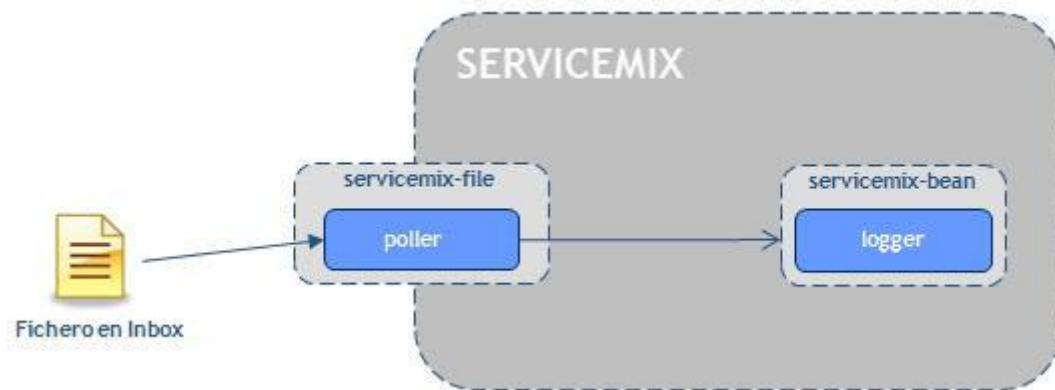


Figura 22. Apache ServiceMix

Como muestra el diagrama de flujo, deberemos crear dos endpoints desplegados sobre los componentes `servicemix-file` y `servicemix-bean` que implementen los servicios a los que están destinados. ServiceMix permite desplegar más de un endpoint por servicio, relación similar a la existente entre puertos e IPs en el caso de direcciones de red, pero con el objetivo de no complicar el tutorial más de lo necesario, mantendremos la relación más básica, es decir, un solo endpoint por servicio.

ServiceMix permite dos modos de despliegue distintos dependiendo de la configuración elegida para los endpoints de la solución de integración. El modo habitual es desplegar ensamblados de service units (cada uno de estos es un conjunto de endpoints que vivirán sobre el mismo componente JBI, configurados en un fichero llamado `xbean.xml`) sobre una instancia previamente levantada de ServiceMix. A cada uno de estos ensamblados se le conoce como service assembly. No obstante, para nuestro tutorial vemos más práctico el usar el modo embebido, ya que facilita enormemente las tareas sobre todo en el ámbito de desarrollo. Esto es debido a que se usa un único fichero de configuración (`servicemix.xml`) para configurar tanto el contenedor JBI como los endpoints que se desplegarán sobre él. Este tipo de despliegue nos permite centrarnos en la creación de los endpoints, además de ahorrarnos el tener que descargar la distribución completa de ServiceMix.

2.3 INDUSTRIA TEXTIL

2.3.1 DEFINICIÓN.

Industria textil es el nombre que se da al sector de la economía dedicado a la producción de ropa, tela, hilo, fibra y productos relacionados. Aunque desde el punto de vista técnico es un sector diferente, en las estadísticas económicas se suele incluir la industria del calzado como parte de la industria textil.

Los textiles son productos de consumo masivo que se venden en grandes cantidades. La industria textil genera gran cantidad de empleos directos e indirectos, tiene un peso importante en la economía mundial. Es uno de los sectores industriales que más controversias genera, especialmente en la definición de tratados comerciales internacionales. Debido principalmente a su efecto sobre las tasas de empleo.

La industria textil y confecciones abarca diversas actividades que van desde el tratamiento de las fibras textiles para la elaboración de hilos, hasta la confección de prendas de vestir y otros artículos. En Lima Metropolitana (Perú), las empresas dedicadas a dichas actividades integran diferentes procesos productivos, lo cual otorga un mayor nivel agregado a sus productos.

La industria textil y confecciones presenta una tendencia creciente en sus niveles productivos apoyada en las confecciones de prendas de vestir, que son enviadas a los mercados nacionales e internacionales, que dada la calidad de las prendas peruanas, son demandadas por el segmento alto del mercado.

La fina tradición textil en el Perú data de tiempos preincaicos y se sustenta en la alta calidad de los insumos utilizados, como la fibra de alpaca y el algodón Pima. La producción textil y de confecciones ha evolucionado en técnica y en diseños convirtiéndose en uno de los productos mejor cotizados en sus respectivas categorías a nivel internacional.

El desarrollo textil en el Perú ha sido posible gracias al cultivo de fibras de excepcional calidad como el algodón Pima peruano (fibra extra larga) y el algodón Tangüis (fibra larga). Además, el Perú es el mayor productor mundial de las fibras más finas de camélidos sudamericanos, entre las que se distinguen las de alpaca y vicuña, siendo esta última superior a la fibra de Cachemira.

La calidad de la confección también resalta tanto por el esfuerzo permanente de los empresarios del sector para mantenerse actualizados tecnológicamente, como por la habilidad y responsabilidad del operador peruano. Así, el Perú ha logrado un buen posicionamiento como proveedor confiable y con un sector textil bien desarrollado, que va desde el cultivo de algodón de indiscutible calidad, y la crianza y esquila de vicuñas y alpacas, a la hilandería, el tejido, teñido y acabado de las telas, y la confección de prendas, permitiendo atender rápidamente pedidos completos de marcas líderes en el ámbito mundial.

En la actualidad para elaborar estas prendas se utiliza el algodón Pima de fibra y pelos finos de alpaca; que pasan por un largo proceso de limpieza y unión de los tejidos, Así con este material terminado se elaboran las prendas de vestir que usamos en la actualidad.

2.3.2 INSTITUCIONES Y REDES DE APOYO A LA PRODUCCIÓN TEXTIL

La actividad textil y de confecciones se encuentra apoyada por múltiples instituciones públicas y privadas que buscan ampliar la participación de productos peruanos en diferentes mercados del mundo, mejorar los procesos productivos y desarrollar redes de proveedores para los diferentes servicios.

Cabe destacar el rol del Instituto Peruano del Algodón que presenta intereses avances respecto de la mejora genética de algodón de fibra larga para sembrar en diversos territorios peruanos. Asimismo, existen diversas entidades públicas y privadas que están apoyando la crianza adecuada y mejoramiento genético gradual de las fibras de auquénidos sudamericanos, así como el desarrollo de hilos y mezclas especiales.

Entre estas tenemos:

Instituciones Públicas:

- PROMPEX: Comisión para la promoción de las exportaciones.
- CONACS: Consejo Nacional de Camélidos Sudamericanos.
- INIA: Instituto Nacional de Investigación y Extensión Agraria.
- MINAG: Ministerio de Agricultura.
- TEXTILE PERU MARKETPLACE
- CAMELYDA: Red de Información en Camélidos y desarrollo de los Andes

Instituciones Privadas:

- ADEX: Asociación de Exportadores del Perú
- COMES: Sociedad de Comercio Exterior del Perú
- SIN: Comité Textil
- IPAC: Instituto Peruano de la Alpaca Y Camélidos
- IPA: Instituto Peruano del Algodón

2.3.3 ELABORACIÓN DE PRODUCTOS TEXTILES

Proceso de producción de fibras naturales

1° etapa: Batan.

En esta etapa se realiza el proceso de abrir, mezclar y limpiar las fibras textiles. Para ello los fardos originales son desmenuzados con el Batán, máquina que permite limpiar las fibras del polvo y demás desperdicios que tenga.

2° etapa: Cardado.

Las fibras son transportadas a la máquina de cardar donde las fibras que forman un conjunto compacto son separadas formando mechas (cardadas o peinadas) y a la vez los rodillos de la máquina continúa limpiando del material.

3° etapa: Manuar.

Este proceso consiste en estirar las mechas formando una cinta gruesa. Al estirar estas mechas se continúa limpiando el algodón y se van formando hebras.

4° etapa: Hilado.

Las hebras pasan a la máquina llamadas continuas. Las hebras delgadas pasan por unos rodillos que retuercen las hebras a la vez que muelen el polvo que todavía se encuentra, y pasa seguidamente a uno conos donde se encona el hilo.

Finalmente el hilo enconado pasa a la etapa pretelar donde preparan el hilo en las máquinas telares. Las empresas que no fabrican la tela embolsan los conos y los lleva al almacén de productos terminados.

2.4.4 IMPORTANCIA DE LA PRODUCCIÓN TEXTIL

En el Perú, las empresas dedicadas a la producción textil están integradas a lo largo del proceso productivo, convirtiéndose en una ventaja competitiva, junto a la calidad de las fibras peruanas que están consideradas entre las mejores del mundo (algodón, lana de alpaca y vicuña) y la cercanía de su principal mercado(EE.UU.); Esto ha permitido a la industria textil y de confecciones presentar una tendencia creciente en su producción. Siendo las prendas de vestir uno de los principales productos, cuya importancia radica en el mayor valor agregado que presenta. Asimismo, el incremento en la producción a llevado a una mayor utilización de su capacidad instalada, fabricación de productos textiles (54.9%) y de prendas textiles (94.2%), siendo superiores al total de la industria no primaria (51.55%).

El ingreso monetario al Perú otorgado por las industrias, al exportar su producción textil es importante ya que; en consecuencia a la elaboración de estas materias primas se entrega al estado un tanto por ciento de las ganancias ya sea por el IGV o pago al exportar los bienes realizados. La economía del Perú actualmente se mantiene y no progresa, por tanto al exportar la producción textil esta generara ganancias tanto a la industria como al país en el general.

Por otro lado, el sector presenta problemas que afectan su desarrollo, como es su alto nivel de endeudamiento que dificulta la obtención de préstamos de largo plazo para la modernización de maquinarias, afectando principalmente a las PYMES en su necesidad de modernizarse, otro problema es el abastecimiento del algodón ocasionado por el rápido crecimiento del sector.

Las empresas en el ámbito de producción textil han crecido exponencialmente dentro de un mercado informal en donde han obtenido una evolución y desarrollo que superó todas las expectativas, convirtiéndose en todo un dinamismo comercial, que no ha sido paralelamente acompañado de un desarrollo estratégico empresarial, sino por el contrario, estas empresas son guiadas empíricamente por empresarios quienes desarrollaron sus habilidades en el día a día, basados en la experiencia y la educación; Asimismo, dado que los mercados globales tienen diferentes demandas y éstas a la vez son extremadamente cambiantes, se debe lograr la diversidad y satisfacer a las demandas producidas por los nuevos y flexibles mercados.

Para lograr esta diversidad se requiere que reorganicen los procesos y busquen aun mayor excelencia en la mano d obra de la producción textil en el Perú.

Existe entonces un gran potencial de desarrollo en las empresas de confecciones actuales y que se han convertido en frentes de producción importantes del mercado como las empresas de “Gamarra”, considerando la necesidad de desarrollar nuevas formas productivas y empresariales que las impulsen y las vuelvan competitivas en el entorno que se les presenta en la actualidad.

De no tomarse en cuenta estos aspectos, la oportunidad creada no será aprovechada y los clientes simplemente buscarán nuevos canales que les permitan satisfacer sus expectativas y probablemente descarten a los proveedores que no respalden sus objetivos de competitividad global, es decir, serán desplazados.

CAPÍTULO III

3. ESTADO DEL ARTE

3.1 CASOS DE ESTUDIO:

Existen diferentes estudios e investigaciones realizadas sobre la integración de sistemas informáticos para la transmisión de información entre diversas plataformas del negocio, en el apartado siguiente mencionaremos y describiremos alguno de ellos a fin de tener una visión más clara de nuestra investigación.

3.1.1. INTEROPERABILIDAD E INTEGRACION DE SISTEMAS INFORMATICOS DE LA IGLESIA CATOLICA EN CHILE

[David Castro Salinas, Tesis de grado, Ing. Informático, Universidad tecnológica Metropolitana Escuela de Informática, Chile, 2007]

Problema:

El Pontificio Consejo de la Iglesia Católica en Chile requiere un medio directo, inmediato, interactivo y participativo como instrumento para su uso administrativo, pues necesitan acceder a múltiples fuentes de datos.

Solución de la Tesis:

En esta tesis se desarrolla la construcción de un sistema integrador, mediante el uso de las tecnologías XML junto a las arquitecturas orientadas a servicios (SOA) para la estructuración y comunicación entre sistemas y bases de datos heterogéneos.

Se plantea la construcción de módulos interoperables para la integración de los sistemas existentes en la Conferencia Episcopal de Chile, de tal forma que éstos en base a parámetros del sistema puedan relacionarse entre sí, y que además permita, la integración de nuevas fuentes de recursos.

3.1.2 PROYECTO DE TESIS: INTEGRACIÓN DE SERVICIOS Y AGENTES DE USUARIO EN LA RECUPERACIÓN DE INFORMACIÓN EN UNA BIBLIOTECA DIGITAL.

[Jorge Cabrera Diaz, Tesis de Grado, Ingeniería en Sistemas Computacionales, Universidad de las Américas-Puebla, México, 2010]

Problema:

Entre los numerosos problemas que se pueden presentar en el manejo y estructuración de una biblioteca digital están los de integración de sus servicios, es por esto que surge la idea de proponer y desarrollar una arquitectura que resuelva este el manejo y recuperación de información de una biblioteca digital.

Solución Planteada:

Esta tesis se centra en los temas referentes a bibliotecas digitales y agentes de usuario, proponiendo una solución basados en éstos para resolver el problema de integración de servicios y recuperación de información en una biblioteca digital.

Mobots es una arquitectura para una biblioteca digital florística diseñada para integrar y manejar diversos servicios y agentes a través de una interfaz común. La arquitectura Mobots se enfoca principalmente en el control de agentes que son creados como instancias de clases de agentes registradas en el sistema. Estas instancias pueden ser manipuladas también a través de Mobots al alterar su estado de ejecución.

Como parte del grupo de agentes que ha de conformar los servicios de la biblioteca digital se desarrolló un agente denominado Flora Retrieval Agent (FRA) para recuperación de información. FRA hace uso del "conocimiento" que tiene sobre las necesidades del usuario, para realizar búsquedas y recuperación de información de manera semi-autónoma -dicho "conocimiento" está almacenado en archivos de preferencias de los usuarios. Otras de sus características importantes es la notificación al usuario de nueva información en el momento en que ésta llega a la biblioteca, evitando con esto horas invertidas y perdidas del usuario en búsquedas sin sentido.

3.1.3 TESIS DOCTORAL: METODOLOGÍA, ESTRUCTURA Y DESARROLLO DE INTERFACES INTERMEDIAS PARA LA CONEXIÓN DE LABORATORIOS REMOTOS Y VIRTUALES A PLATAFORMAS EDUCATIVAS

[Elio San Cristóbal Ruiz, Tesis Doctoral, Ingeniería en Informática, UNIVERSIDAD NACIONAL DE EDUCACIÓN A DISTANCIA, España, 2010]

Problema:

El proceso de aprendizaje implica, en muchos casos, la unión de dos tipos de conocimiento: un conocimiento teórico y una aplicación práctica de dicho conocimiento. Hasta hace unos años, ambos conocimientos eran adquiridos de forma presencial: en aulas de aprendizaje y laboratorios físicos. Actualmente, el desarrollo de las redes comunicaciones y la aparición de Internet ha impulsado otras metodologías de enseñanza, como la enseñanza mixta o a distancia. Como consecuencia, las aulas y los laboratorios han pasado a ser virtuales o remotos.

Solución Planteada:

Esta Tesis está motivada por la necesidad de crear una arquitectura o middleware capaz de unir ambas soluciones en una sola y por tanto permitir:

- Que un estudiante pueda utilizar una única solución para adquirir conocimiento teórico y práctico.
- Reutilizar los servicios ofrecidos por las plataformas educativas. De tal forma que los diseñadores del laboratorio se puedan simplemente limitar a diseñar y definir los experimentos más adecuados a cada laboratorio.
- Utilizar los estándares e-learning soportados por las plataformas educativas. Y por tanto reutilizar cursos, capítulos o elementos de evaluación realizados por otras instituciones.

- Reutilizar laboratorios ya existente, dándole la posibilidad a una institución de no tener que desarrollar el mismo laboratorio, con el gasto que ello supone: económico, personal, etc.
- Crear una arquitectura de conexión global de laboratorios y sistemas de gestión de aprendizaje de diferentes universidades e instituciones. Como una red de servicios global.

3.2 SELECCIÓN Y JUSTIFICACIÓN DE LA ARQUITECTURA PARA LA SOLUCIÓN

3.2.1 ARQUITECTURAS A EVALUAR:

Los diseños posibles a emplear para la solución del problema planteado en la tesis son:

1. Arquitectura Orientado a Servicios(SOA):
2. Patrones de Integración (EAI)

3.2.2 CRITERIOS DE EVALUACIÓN DE LA ARQUITECTURA:

- a. Aporta Rentabilidad
- b. Reutilizable
- c. Soluciones de bajo tiempo de espera.
- d. Permite integración de bajo costo.
- e. Integración a gran escala a largo plazo.

3.2.3 ANÁLISIS COMPARATIVO DE ARQUITECTURAS:

El rango de evaluación será la siguiente:

Cuadro 1. Rango de evaluación de Arquitectura	1	Muy bajo
	2	Bajo
	3	Regular
	4	Alto
	5	Muy alto

Modelo Criterios	SOA	EAI
a. Aporta Rentabilidad	5	5
b. Reutilizable	4	4
c. Soluciones de bajo tiempo de espera.	5	4
d. Permite integración de bajo costo.	3	3
e. Integración a gran escala a largo plazo.	5	4
TOTAL	22	20

Cuadro 2. Cuadro Comparativo de Arquitecturas

Artículos:

Giga "Target Business Process Improvements for Best Return on Application Integration Spending."

Resultado:

La Arquitectura Orientada a Servicios (SOA, Service Oriented Architecture) supone una estrategia

general de organización de los elementos de IT, de forma que una colección abigarrada de sistemas distribuidos y aplicaciones complejas se pueda transformar en una red de recursos integrados, simplificada y sumamente flexible. Un proyecto SOA bien ejecutado permite alinear los recursos de IT de forma más directa con los objetivos de negocio, ganando así un mayor grado de integración con clientes y proveedores, proporcionando una inteligencia de negocio más precisa y más accesible con la cual se podrán adoptar mejores decisiones, y ayuda a las empresas a optimizar sus procesos internos y sus flujos de información para mejorar la productividad individual. El resultado neto es un aumento muy notable de la agilidad de la organización.

3.3. SELECCIÓN Y JUSTIFICACIÓN DE LA HERRAMIENTA PARA LA SOLUCIÓN

3.3.1 HERRAMIENTAS A EVALUAR:

Las herramientas son las siguientes:

1. Apache Service Mix
2. Oracle Service Bus
3. Enterprise Service Bus

3.3.2 CRITERIOS DE EVALUACIÓN DE LA HERRAMIENTA:

- a) Soporte tecnológico.
- b) Sistemas operativos en que puede ejecutarse.
- c) Complejidad de la herramienta.
- d) Costos de implementación.
- e) Performance.
- f) Fiabilidad.
- g) Documentación.
- h) Orquestación de Servicios.
- i) Integración con otras tecnologías (ejb, Spring, etc).

3.3.3 ANÁLISIS COMPARATIVO DE HERRAMIENTAS:

El rango de evaluación será la siguiente:

Cuadro 3. Rango de evaluación de Herramientas	1	Muy bajo
	2	Bajo
	3	Regular
	4	Alto
	5	Muy alto

Herramienta Criterios	Apache ServiceMix	OSB	ESB
a.	3	5	5
b.	5	5	5
c.	4	4	3
d.	5	3	1
e.	4	5	5
f.	3	5	5
g.	3	5	5
h.	4	4	4
i.	5	5	5
TOTAL	36	41	38

Cuadro 4. Cuadro Comparativo de Herramientas

ARTÍCULO:

The Forrester Wave™: Enterprise Service Bus, by Ken Vollmer for Application Development & Delivery Professionals.

Resultado:

De acuerdo a la evaluación que se realizó a las diferentes herramientas, se decide optar por implementar la solución con Oracle Service Bus (OSB), pues principalmente es la herramienta licenciada con menor costo y con soporte tecnológico asegurado que brinda confiabilidad a sus usuarios. También es importante mencionar la abundante documentación que ayuda a los desarrolladores a comprender mejor la herramienta y hacer un mejor uso de la misma.

IV. APORTE TEÓRICO

4.1 ANALISIS DE IMPACTO

PROCESOS DEL NEGOCIO

- 1.- Proceso de Ventas
- 2.- Proceso de Almacenaje y Stock
- 3.- Proceso de Producción
- 4.- Proceso de Compras
- 5.- Proceso de Distribución

ROLES DEL ENTORNO DE NEGOCIO:

Actores del negocio

Cliente

Proveedor

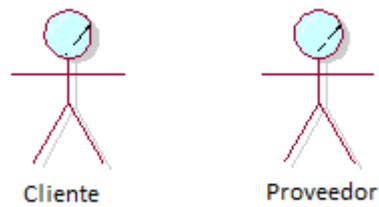


Figura 23. Roles Entorno de Negocio

Trabajadores del negocio

Encargado de Ventas

Encargado de Compras

Encargado de Almacén

Encargado de Distribución

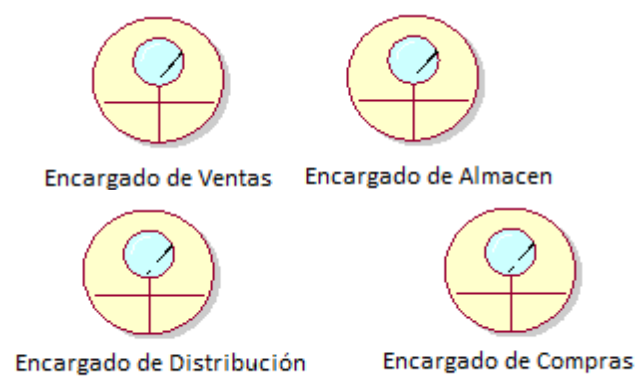
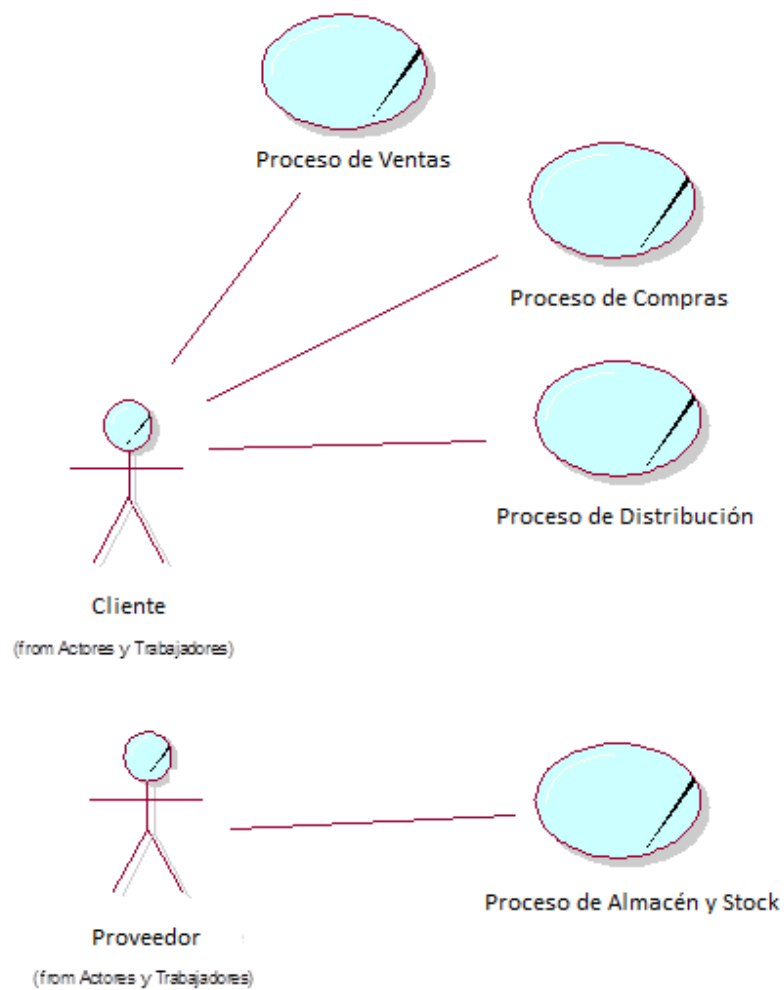


Figura 24. Trabajadores del Negocio

4.2 MODELO DE CASOS DE USO DEL NEGOCIO:

Con Actores:



Propios:



Figura 25. Modelo Casos de Uso

DESCRIPCION DEL PROCESO DE NEGOCIO:

Nombre del CUN 1: Proceso de Ventas

Flujo normal:

- 1 El CUN comienza cuando el cliente entra en contacto con el Encargado de Ventas.
- 2 El Encargado de Ventas recepciona al cliente.
- 3 El cliente solicita información de ciertos productos establecidos.
- 4 Encargado de Ventas muestra su catálogo de productos.
- 5 Cliente indica sus necesidades y requerimientos.
- 6 Encargado de Ventas muestra soluciones al cliente.
- 7 Cliente requiere cotización.
- 8 Encargado de Ventas realiza cotización.
 - 8.1 Si el cliente queda satisfecho se sigue con el paso 9.
 - 8.2 Si el cliente no se encuentra satisfecho, el CUN finaliza.
- 9 El encargado de Ventas revisa el stock en la tienda.
 - 9.1 Si existe stock se pasa al paso 10.
 - 9.2 Si no existe stock se consulta con almacén encargado de ventas y se envía la orden de venta al encargado de almacén.
- 10 El cliente y el Encargado de Ventas llegan a un acuerdo.
- 11 El Encargado de Ventas realiza el contrato.
- 12 El cliente revisa y valida el contrato.
 - 12.1 Si el cliente está de acuerdo firma el contrato y se sigue con el paso 13.
 - 12.2 Si el cliente no está de acuerdo se vuelve al paso 10.
- 13 El encargado de ventas firma el contrato.
- 14 El encargado de ventas genera la orden de venta.
- 15 El CUN finaliza.

USUARIOS IMPLICADOS EN EL PROCESO DE NEGOCIO: Proceso de Ventas.

Actor del negocio:

- Cliente



Trabajadores del proceso

- Encargado de Ventas



ACCIONES NECESARIAS PARA REALIZAR EL PROCESO DEL NEGOCIO:

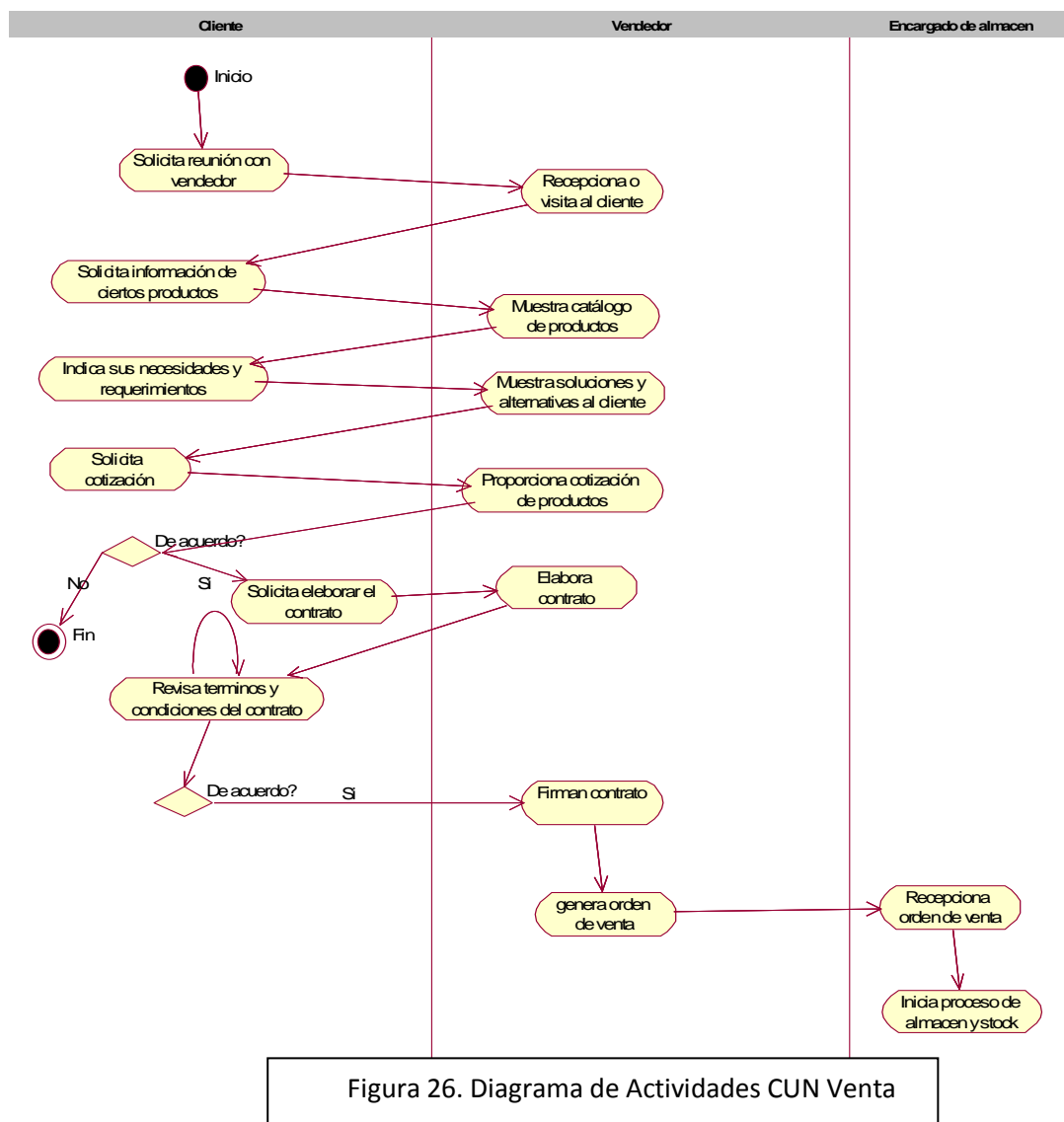
. Cliente:

- Solicita información
- Indica sus necesidades y requerimientos
- Solicita cotizaciones
- Revisa el contrato
- Acepta y firma el contrato

. Encargado de Ventas:

- Proporciona información.
- Presenta soluciones posibles a las necesidades del cliente.
- Realiza cotización
- Realiza contrato
- Firma de contrato
- Envía orden de venta

DIAGRAMA DE ACTIVIDADES DEL PROCESO DE NEGOCIO



LISTA DE ACTIVIDADES:

- Iniciación de contacto entre cliente y el encargado de ventas.
- Recepción del cliente
- Solicitud de catálogo de productos
- Indicaciones sobre las necesidades y requerimientos
- Solicitud de cotizaciones
- Revisión de contrato
- Aceptación y firma de contrato
- Realización de cotizaciones
- Realización de contrato
- Envío de orden de venta

LISTA DE ENTIDADES DEL PROCESO DE NEGOCIO: Proceso de Ventas.

- Catálogo de productos
- Contrato
- Orden de Venta.

DESCRIPCION DEL PROCESO DE NEGOCIO:

Nombre del CUN 2: Proceso de Almacén y Stock

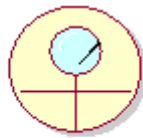
Flujo normal:

- 1 El CUN comienza con la llegada de la orden de Venta.
- 2 El Encargado de Almacén revisa Stock.
 - 2.1 Si hay stock de pedidos de acuerdo a la orden de venta, se continúa con el paso 3.
 - 2.2 Si no hay stock se elabora una orden de Producción
 - 2.2.1 El encargado de almacén envía orden de producción al encargado de producción.
 - 2.2.2 Comienza el proceso de producción.
 - 2.2.3 El encargado de almacén recibe orden de producción concluida y se vuelve al paso 2.
- 3 El encargado de almacén empaqueta los pedidos de acuerdo a la orden de venta.
- 4 El encargado de almacén contacta al Encargado de Distribución.
- 5 Se da inicio al proceso de distribución.
- 6 El CUN finaliza.

USUARIOS IMPLICADOS EN EL PROCESO DE NEGOCIO: Proceso de Almacén y Stock

Trabajadores del proceso

- Encargado de Almacén



Encargado de Almacen

ACCIONES NECESARIAS PARA REALIZAR EL PROCESO DEL NEGOCIO:

. Encargado de Almacén:

- Recibe orden de venta.
- Revisa Stock
- Envía orden de producción
- Recibe orden de producción concluida
- Empaca pedidos
- Envía orden de distribución al encargado de distribución

DIAGRAMA DE ACTIVIDADES DEL PROCESO DE NEGOCIO

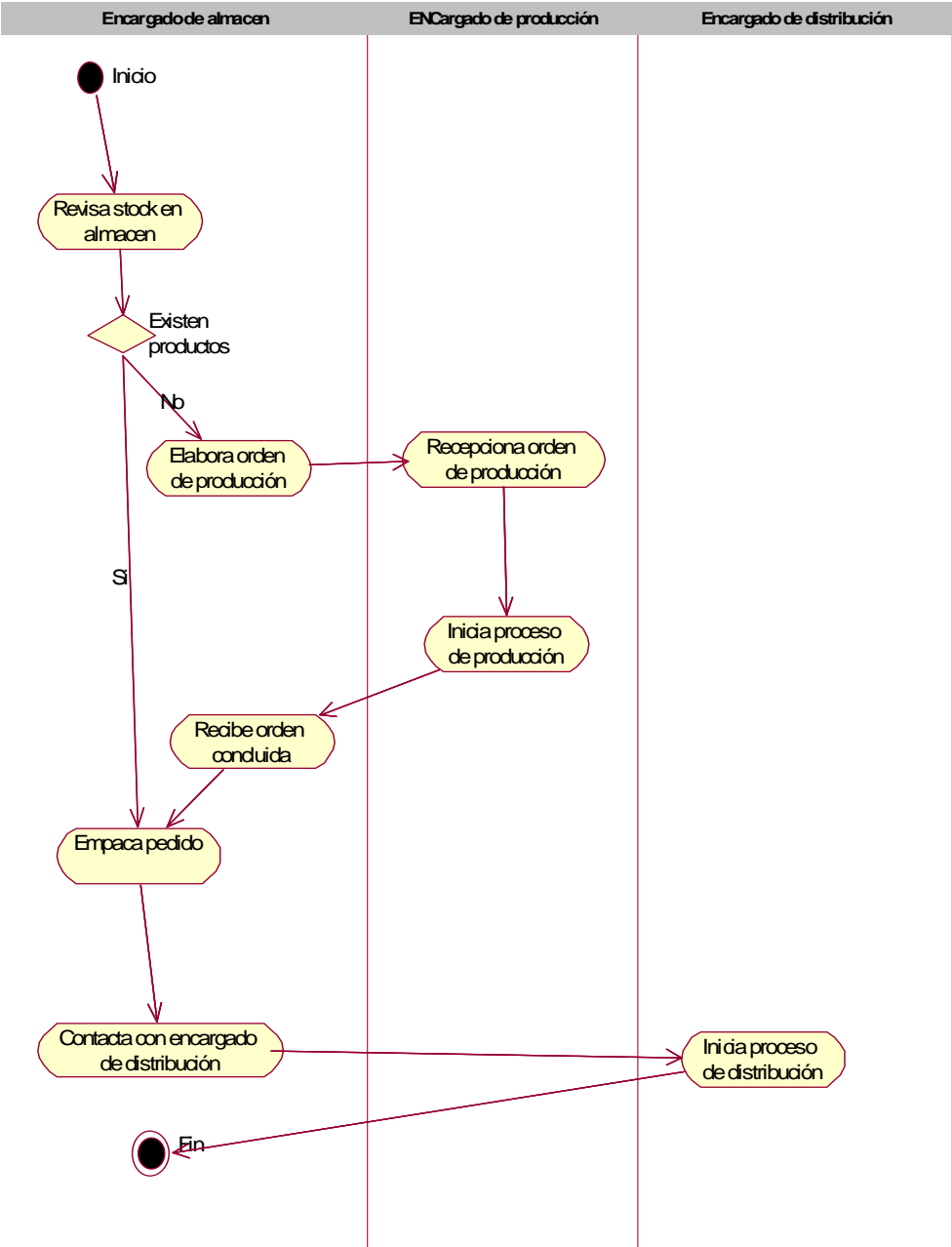


Figura 27. Diagrama de Actividades CUN Almacenaje

LISTA DE ACTIVIDADES:

- Recepción de orden de venta.
- Revisión de Stock
- Envío de la orden de producción
- Recepción de la orden de producción concluida
- Empaque de pedidos
- Envío de orden de distribución.

LISTA DE ENTIDADES DEL PROCESO DE NEGOCIO: Proceso de Almacén y Stock.

- Orden de Venta.
- Orden de producción
- Orden de producción concluida.
- Orden de distribución

DESCRIPCION DEL PROCESO DE NEGOCIO:

Nombre del CUN 3: Proceso de Producción

Flujo normal:

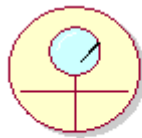
- 1 El CUN comienza cuando el Encargado de producción recibe la orden de producción.
- 2 El Encargado de Producción revisa si hay insumos y herramientas necesarias para la elaboración de tela.
 - 2.1 Si existen insumos y las herramientas necesarias se sigue con el paso 3.
 - 2.2 Si no existen insumos o herramientas necesarias se genera una orden de compra.
 - 2.2.1 Se da inicio al CUN de Compras.
 - 2.2.2 Se retorna al paso 2.
- 3 Encargado de Producción ordena comenzar con la producción.
- 4 Los operarios arman el punto adecuado de tejido y seleccionan los hilos correspondientes a la calidad acordada.
 - 4.1 Si el hilo se rompe, se pausa la producción hasta su correspondiente solución y se vuelve al paso 7.
- 5 Las telas pasan por un control de calidad.
 - 5.1 Si se encuentran fallas leves se manda a retocar inmediatamente.
 - 5.2 Si se encuentran fallas graves, se genera una nueva orden de producción y se vuelve al paso 1.
 - 5.3 Si las telas aprueban el control de calidad se sigue con el paso 6.
- 6 Se llevan los rollos de tela al almacén.

- 7 Encargado de Producción envía orden de producción concluida a Almacén.
- 8 Se da inicio al CUN de Distribución.
- 9 El CUN finaliza.

USUARIOS IMPLICADOS EN EL PROCESO DE NEGOCIO: Proceso de Producción.

Trabajadores del proceso

- Encargado de Producción



Encargado de Producción

ACCIONES NECESARIAS PARA REALIZAR EL PROCESO DEL NEGOCIO:

. Encargado de Producción:

- Recepción de la orden de producción.
- Revisión de insumos y herramientas.
- Generación de la orden de producción concluida
- Generación de una orden de compra.

DIAGRAMA DE ACTIVIDADES DEL PROCESO DE NEGOCIO

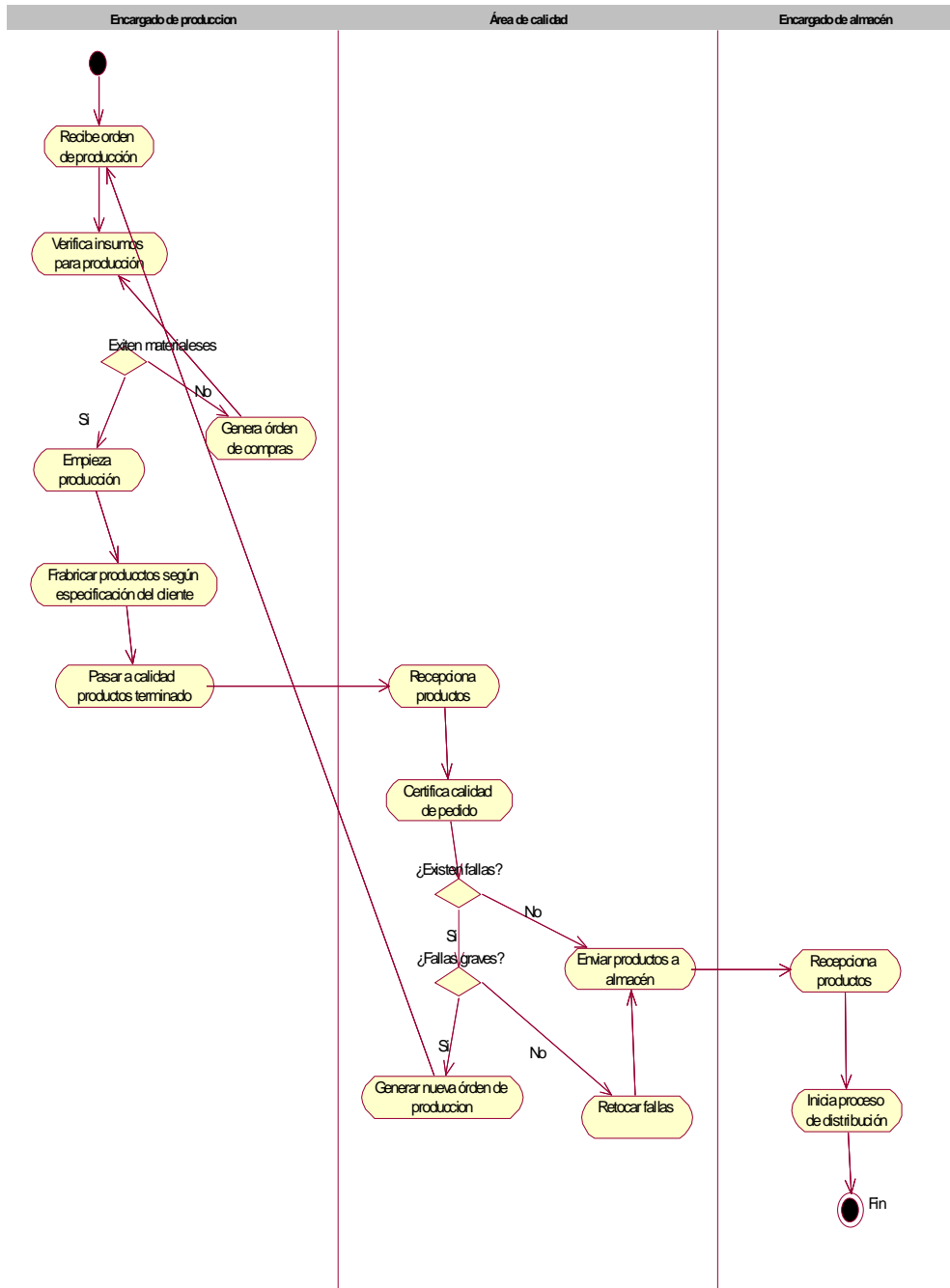


Figura 28. Diagrama de Actividades CUN Producción

LISTA DE ACTIVIDADES:

- Recepción de la orden de producción.
- Revisión de insumos y herramientas
- Generación de la orden de producción concluida
- Generación de orden de compra.

LISTA DE ENTIDADES DEL PROCESO DE NEGOCIO: Proceso de Producción.

- Orden de producción
- Orden de producción concluida
- Orden de compra.

DESCRIPCION DEL PROCESO DE NEGOCIO:

Nombre del CUN 4: Proceso de Compra.

Flujo normal:

- 1 El CUN comienza cuando el Encargado de Compras recibe la orden de compra.
- 2 El Encargado de Compras revisa los proveedores establecidos por cada recurso a comprar.
- 3 El Encargado de Compras selecciona los proveedores.
- 4 El encargado de compras se contacta con los proveedores asignados.
- 5 El encargado de compras genera la lista de compras por cada proveedor establecido.
- 6 El encargado de compras revisa la lista de compras al momento de recibir los pedidos y recursos adquiridos.
 - 6.1 Si falta algún recurso se coordina con el proveedor para una nueva entrega y se continua con el paso 5.
- 7 El CUN finaliza.

USUARIOS IMPLICADOS EN EL PROCESO DE NEGOCIO: Proceso de Compra.

Actor del negocio:

- Proveedor



Trabajadores del proceso

- Encargado de Compras



ACCIONES NECESARIAS PARA REALIZAR EL PROCESO DEL NEGOCIO:

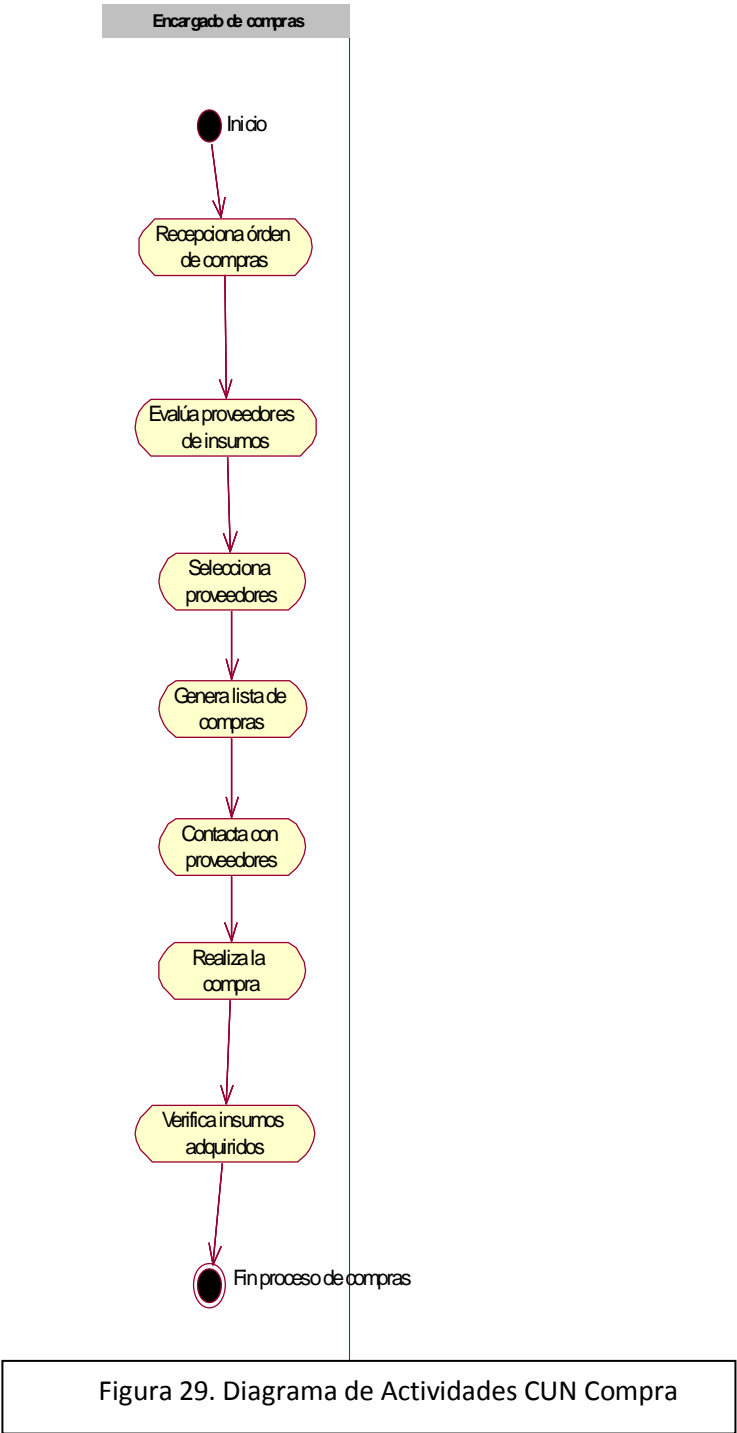
. Proveedor:

- Entrega de productos comprados.

. Encargado de Compras:

- Recepciona la orden de compra.
- Selecciona proveedores
- Genera la lista de compras.

DIAGRAMA DE ACTIVIDADES DEL PROCESO DE NEGOCIO



LISTA DE ACTIVIDADES:

- Iniciación de contacto entre cliente y el encargado de ventas.
- Recepción del cliente
- Solicitud de catálogo de productos
- Indicaciones sobre las necesidades y requerimientos
- Solicitud de cotizaciones
- Revisión de contrato
- Aceptación y firma de contrato
- Realización de cotizaciones
- Realización de contrato
- Envío de orden de venta

LISTA DE ENTIDADES DEL PROCESO DE NEGOCIO: Proceso de Compra.

- Orden de compra.
- Lista de compras.

DESCRIPCION DEL PROCESO DE NEGOCIO:

Nombre del CUN 5: Proceso de Distribución.

Flujo normal:

- 1 El CUN comienza cuando el Encargado de Distribución recibe la orden de distribución.
- 2 El encargado de distribución hace entrega de los pedidos vendidos.
- 3 Se finaliza el CUN.

USUARIOS IMPLICADOS EN EL PROCESO DE NEGOCIO: Proceso de Distribución.

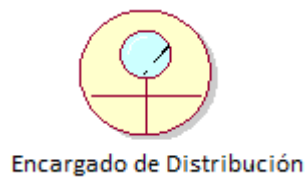
Actor del negocio:

- Cliente



Trabajadores del proceso

- Encargado de Distribución



ACCIONES NECESARIAS PARA REALIZAR EL PROCESO DEL NEGOCIO:

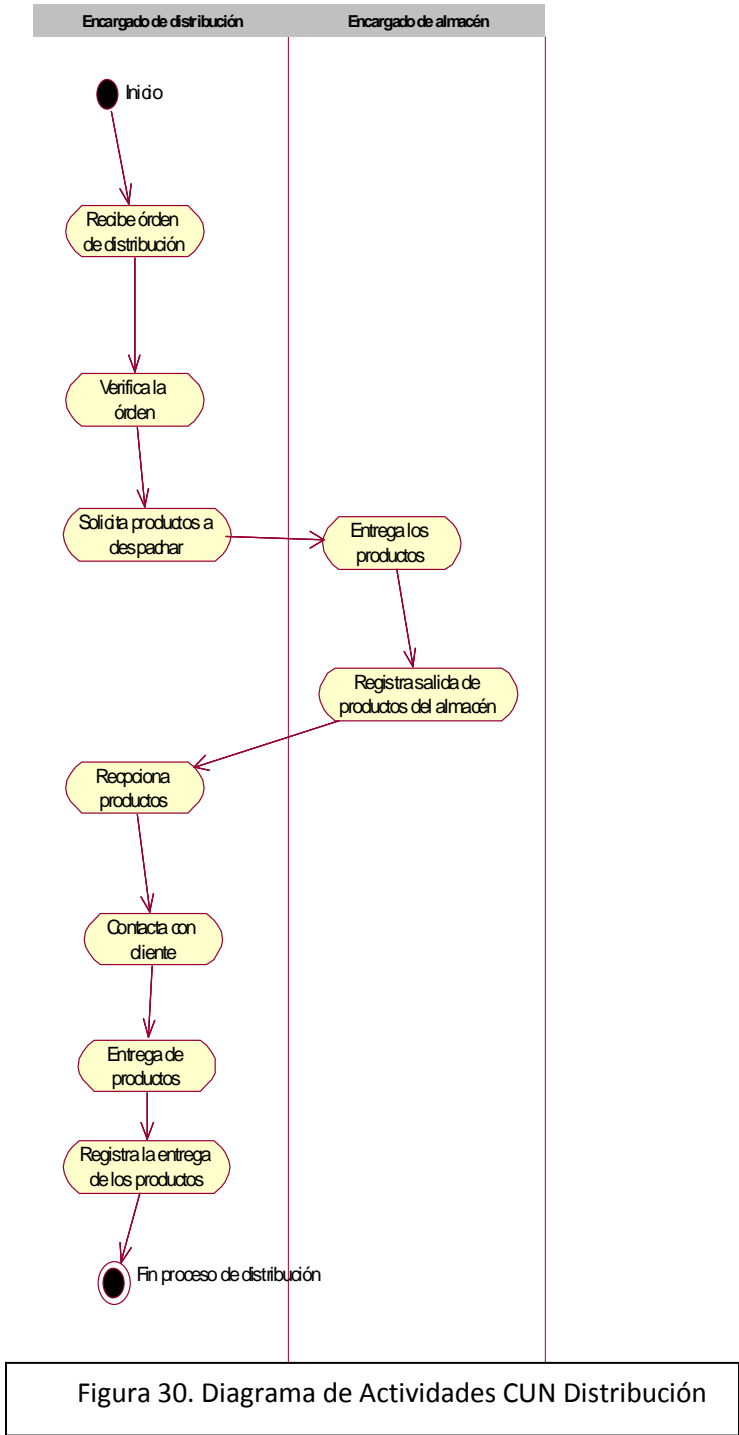
. Cliente:

- Recibe al encargado de distribución
- Firma documento de conformidad de entrega.

. Encargado de Distribución:

- Se dirige donde el cliente.
- Entrega de productos al cliente.

DIAGRAMA DE ACTIVIDADES DEL PROCESO DE NEGOCIO



LISTA DE ACTIVIDADES:

- Apersonarse donde el cliente.
- Firma de documento de conformidad de entrega.

LISTA DE ENTIDADES DEL PROCESO DE NEGOCIO: Proceso de Distribución.

- Orden de distribución
- Documento de conformidad de entrega.

4.3 CASOS DE USO DE SISTEMA

ESPECIFICACIONES DE CASOS DE USO DEL SISTEMA DE LOS PROCESOS DE NEGOCIO:

GESTION DE COMPRAS

Caso de Uso	<i>Registrar Compras</i>
Objetivo	Realizar las compras necesarias por la empresa.
Actor	Encargado de Compras
Precondición	– El encargado de compras recibe la orden de compra.
Pasos	<ol style="list-style-type: none">1. A: El encargado de ventas ingresa al sistema y se loguea.2. S: El sistema muestra la pantalla de bienvenida.3. A: El encargado de ventas consulta los datos de sus proveedores y elige al más conveniente con el cual llegan a un acuerdo.4. A: Registra la posible compra y guarda los datos hasta recibir la conformidad de la entrega.5. S: Guarda los datos de la posible compra.6. A: Una vez que reciba la entrega del proveedor registra la conformidad en el sistema.7. S: Registra la información en su base de datos.
Variaciones	<ol style="list-style-type: none">1. A: El encargado de compras hace clic en “Cancelar” y el CUS finaliza.
Poscondición	Se ha realizado el registro de la compra.
Extensiones	
Requisitos especiales	

Caso de Uso	<i>Consulta de compras realizadas</i>
Objetivo	Realizar la consulta de las compras efectuadas.
Actor	Encargado de Compras
Precondiciones	- Registrar compras en el sistema.
Pasos	<ol style="list-style-type: none"> 1. A: El encargado de compras ingresa al sistema y se loguea. 2. S: Muestra la pantalla de bienvenida. 3. A: Elige "Consultar Compras". 4. S: Muestra pantalla del modulo. 5. A: Ingresa datos para la búsqueda en la consulta. 6. S: Muestra el reporte de consulta. 7. A: La cajera hace click en Finalizar y el CUS finaliza.
Variaciones	<ol style="list-style-type: none"> 1. A: El encargado de Compras hace click en "Cancelar" y el CUS finaliza.
Poscondición	- El sistema ha emitido un reporte de consulta.
Extensiones	
Requisitos especiales	

GESTION DE ALMACENAJE

Caso de Uso	<i>Registro de productos</i>
Objetivo	Registrar nuevos productos en el sistema
Actor	Encargado de almacén
Precondiciones	- Producción envía nuevos productos a almacenar
Pasos	<ol style="list-style-type: none"> 1. A: El encargado de almacén ingresa al sistema y se loguea. 2. S: Muestra la pantalla de bienvenida. 3. A: Elige "Registrar Productos". 4. S: Muestra pantalla del modulo. 5. A: Ingresa datos a registrar y da click en "registrar" 6. S: El sistema valida datos ingresados. 7. A: El encargado da click en Finalizar y el CUS finaliza.
Variaciones	<ol style="list-style-type: none"> 1. A: El encargado de Almacen hace click en "Cancelar" y el CUS finaliza. <p>6.Si los datos no son correctos se vuelve al</p>

	paso 4.
Poscondición	- El sistema ha registrado los productos.
Extensiones	
Requisitos especiales	

Caso de Uso	<i>Consultar Productos</i>
Objetivo	Consultar cantidad de productos.
Actor	Encargado de almacén
Precondiciones	- Se requiere saber la cantidad que se tiene de cierto producto en el almacén.
Pasos	<ol style="list-style-type: none"> 1. A: El encargado de almacén ingresa al sistema y se loguea. 2. S: Muestra la pantalla de bienvenida. 3. A: Elige "Consultar Productos". 4. S: Muestra pantalla del modulo. 5. A: Ingresa datos a Consultar y da click en "consultar" 6. S: El sistema realiza la búsqueda y muestra datos encontrados. 7. A: El encargado da click en Finalizar y el CUS finaliza.
Variaciones	<ol style="list-style-type: none"> 1. A: El encargado de Almacén hace click en "Cancelar" y el CUS finaliza. <p>6.Si los datos no son encontrados se vuelve al paso 4.</p>
Poscondición	- El sistema ha mostrado la cantidad de los productos buscados.
Extensiones	
Requisitos especiales	

Caso de Uso	<i>Registrar Salida de Productos</i>
Objetivo	Registrar el egreso de productos para su control
Actor	Encargado de almacén
Precondiciones	- Se pide la entrega de productos por parte del almacén.

Pasos	<ol style="list-style-type: none"> 1. A: El encargado de almacén ingresa al sistema y se loguea. 2. S: Muestra la pantalla de bienvenida. 3. A: Elige “Registrar salida de productos”. 4. S: Muestra pantalla del modulo. 5. A: Ingresa la información requerida por el sistema para el registro de productos salidos como tipo producto, cantidad y destino. Da click en “registrar” 6. S: El sistema valida la información ingresada. 7. A: El encargado da click en Finalizar y el CUS finaliza.
Variaciones	<ol style="list-style-type: none"> 1. A: El encargado de Almacen hace click en “Cancelar” y el CUS finaliza. 6.Si los datos no son validos se vuelve al paso 4.
Poscondición	- El sistema ha registrado la salida de productos en el almacén.
Extensiones	
Requisitos especiales	

GESTION DE VENTAS

Caso de Uso	<i>Registrar Venta</i>
Objetivo	Registrar productos vendidos
Actor	Encargado de ventas.
Precondiciones	- El cliente solicita productos.
Pasos	<ol style="list-style-type: none"> 1. A: El encargado de ventas ingresa al sistema y se loguea. 2. S: Muestra la pantalla de bienvenida. 3. A: Elige “Registrar venta”. 4. S: Muestra pantalla del modulo. 5. A: Ingresa la información requerida por el sistema para el registro de ventas. Da click en “registrar” 6. S: El sistema valida la información ingresada. 7. S: El sistema genera la orden de venta. 8. A: Imprime la orden y se lo envía a Almacén. 9. A: El encargado da click en Finalizar y el CUS finaliza.
Variaciones	<ol style="list-style-type: none"> 1. A: El encargado de Ventas hace click en “Cancelar” y el CUS finaliza. 6.Si los datos no son validos se vuelve al paso 4.

Poscondición	- El sistema ha registrado la venta.
Extensiones	
Requisitos especiales	

Caso de Uso	<i>Consultar productos</i>
Objetivo	Consultar productos que se tiene en stock en la misma tienda
Actor	Encargado de ventas.
Precondiciones	- El cliente solicita consultar productos.
Pasos	<ol style="list-style-type: none"> 1. A: El encargado de ventas ingresa al sistema y se loguea. 2. S: Muestra la pantalla de bienvenida. 3. A: Elige "Consultar productos". 4. S: Muestra pantalla del modulo. 5. A: Ingresa la información requerida por el sistema para la consulta deseada. Da click en "consultar" 6. S: El sistema valida la información ingresada. 7. S: El sistema muestra resultados de búsqueda. 8. A: El encargado da click en Finalizar y el CUS finaliza.
Variaciones	<ol style="list-style-type: none"> 1. A: El encargado de Ventas hace click en "Cancelar" y el CUS finaliza. <p>6.Si los datos no son validos se vuelve al paso 4.</p>
Poscondición	- El sistema ha realizado la consulta.
Extensiones	
Requisitos especiales	

Caso de Uso	<i>Registrar Nuevo producto en tienda</i>
Objetivo	Registrar nuevos productos ingresados a la tienda.
Actor	Encargado de ventas.
Precondiciones	- La tienda ha recibido nuevos productos para su stock.
Pasos	<ol style="list-style-type: none"> 1. A: El encargado de ventas ingresa al sistema y se loguea. 2. S: Muestra la pantalla de bienvenida. 3. A: Elige "Registrar nuevos productos en tienda". 4. S: Muestra pantalla del modulo. 5. A: Ingresa la información requerida por el sistema

	<p>para el registro. Da click en “registrar”</p> <p>6. S: El sistema valida la información ingresada.</p> <p>7. A: El encargado da click en Finalizar y el CUS finaliza.</p>
Variaciones	<p>1. A: El encargado de Ventas hace click en “Cancelar” y el CUS finaliza.</p> <p>6.Si los datos no son validos se vuelve al paso 4.</p>
Poscondición	- El sistema ha registrado el producto.
Extensiones	
Requisitos especiales	

4.4 DIAGRAMA FUNCIONAL DE NEGOCIO.

Proceso de Compra:

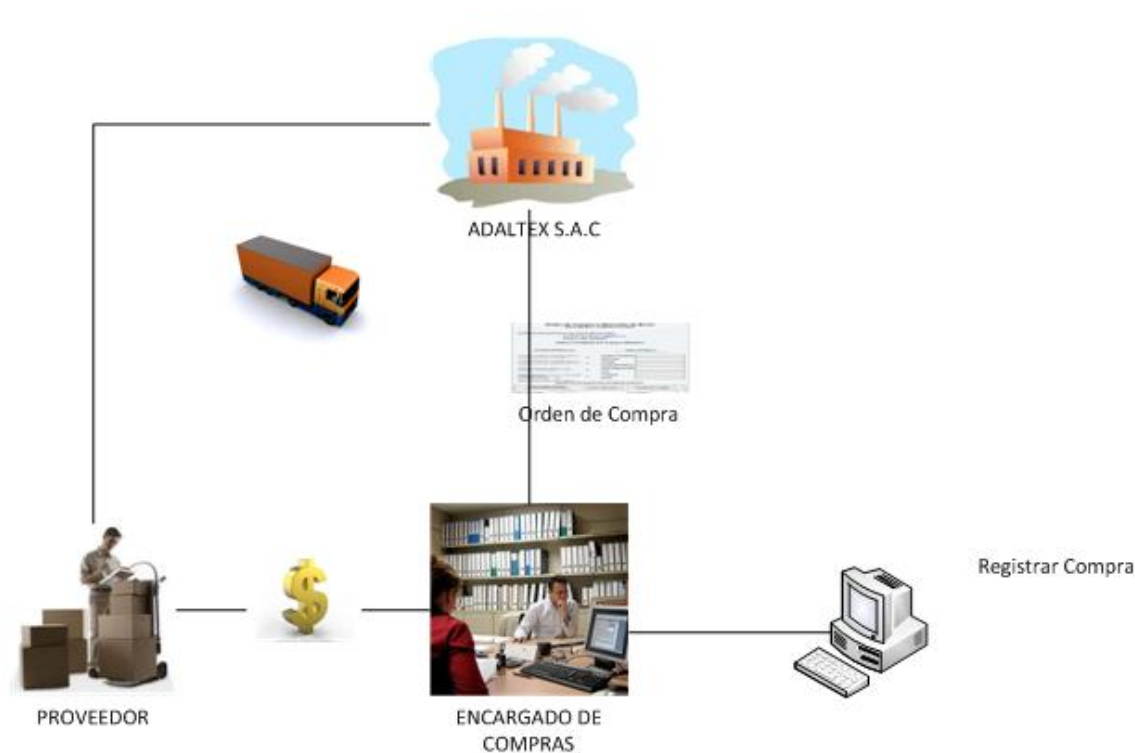


Figura 31. Diagrama Funcional del proceso de Compras

Proceso de Venta:

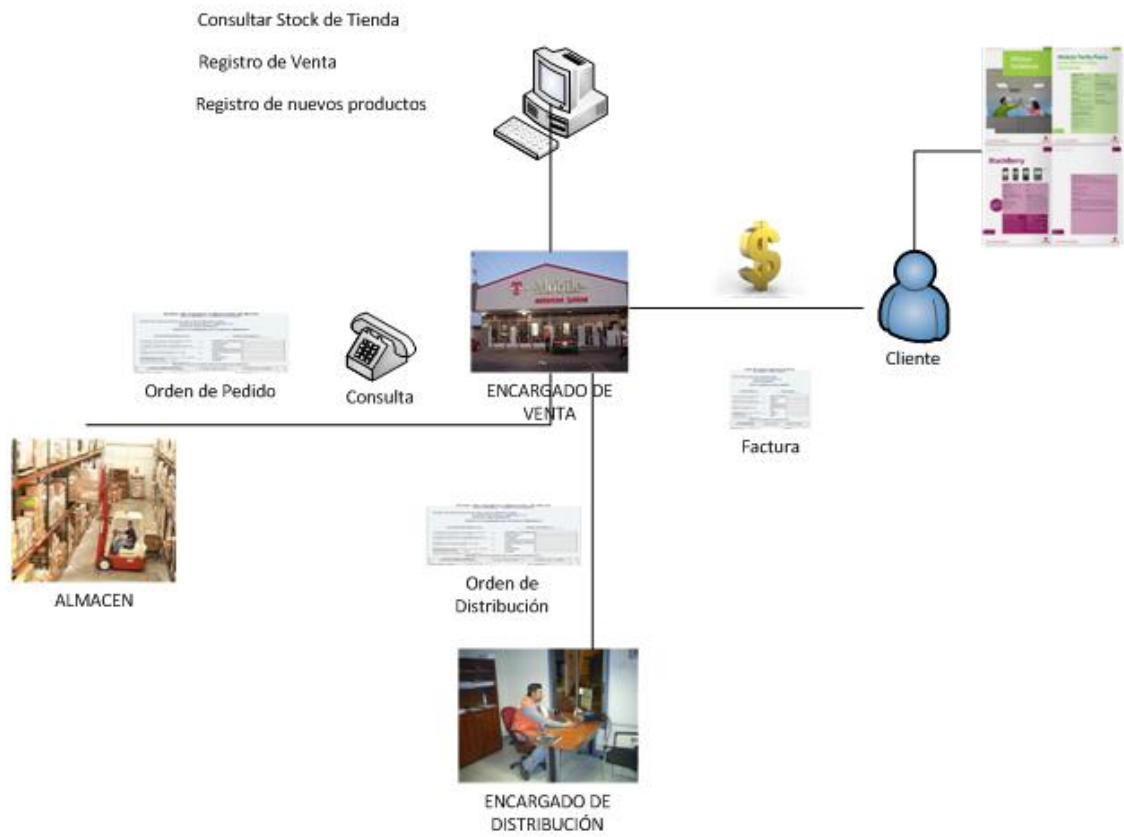


Figura 32. Diagrama Funcional del proceso de Venta

Proceso de Almacenaje:

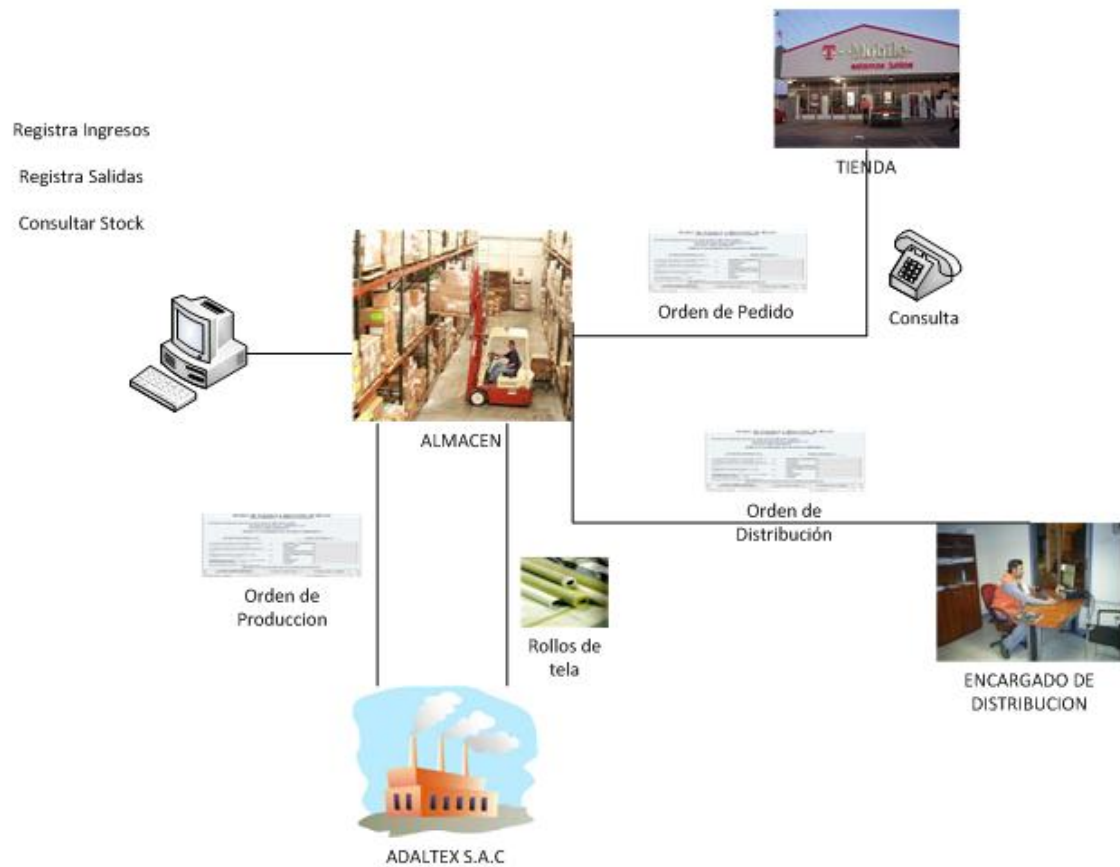


Figura 33. Diagrama Funcional del proceso de Almacén

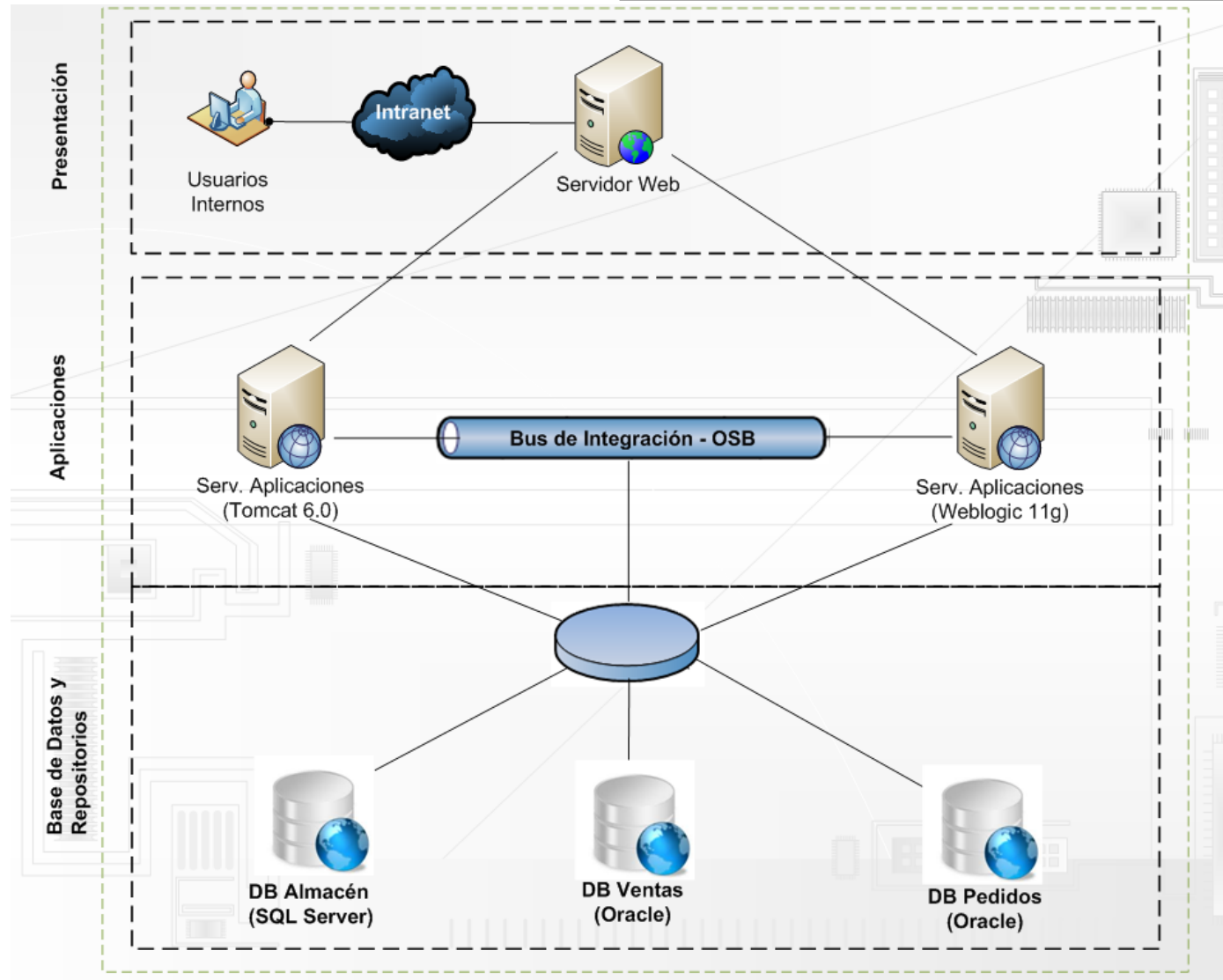
Proceso de Distribución:



Figura 34. Diagrama Funcional del proceso de Distribución

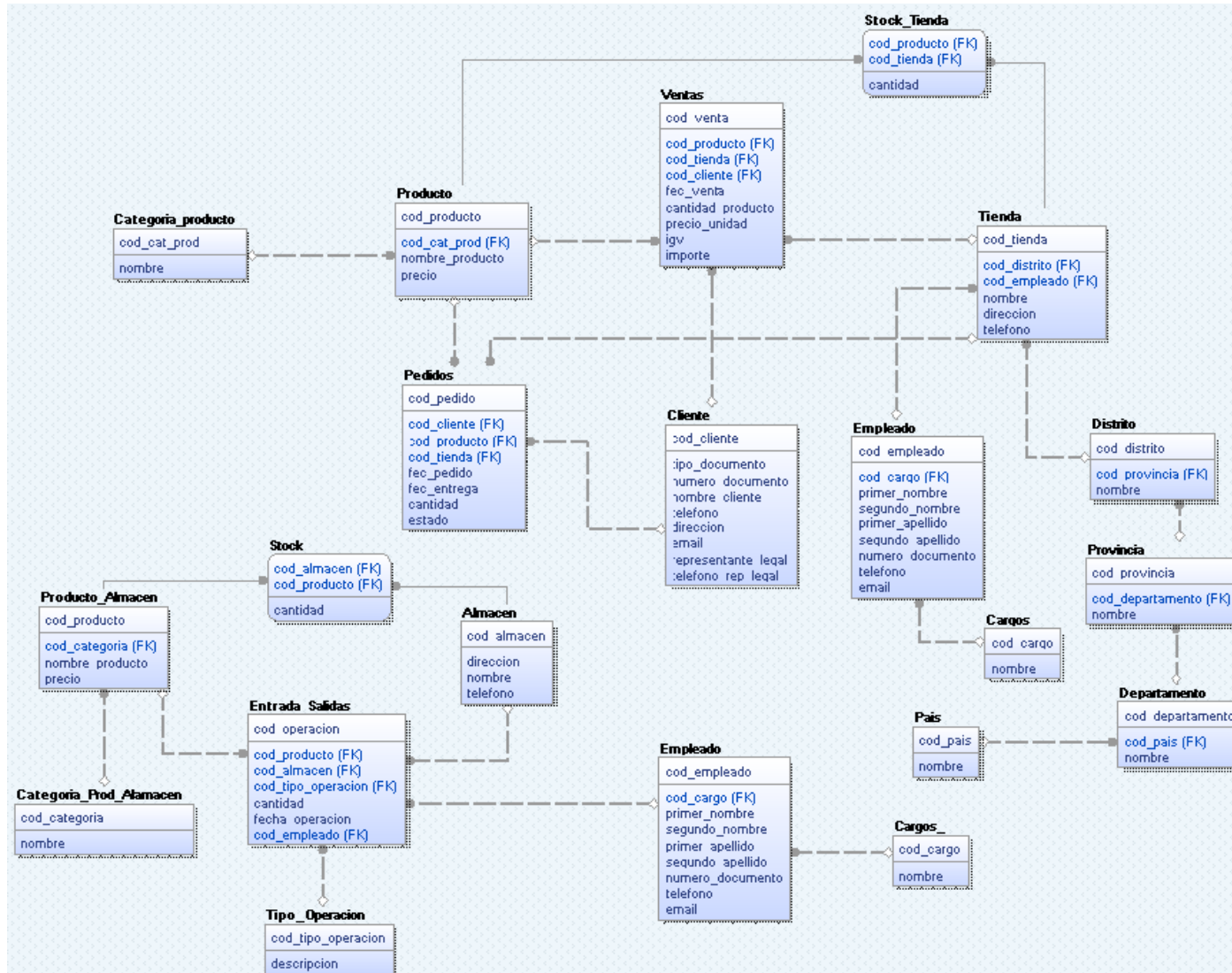
4.5. ARQUITECTURA BUS:

Figura 35. Diagrama Arquitectura Bus



4.6 MODELO DE DATOS

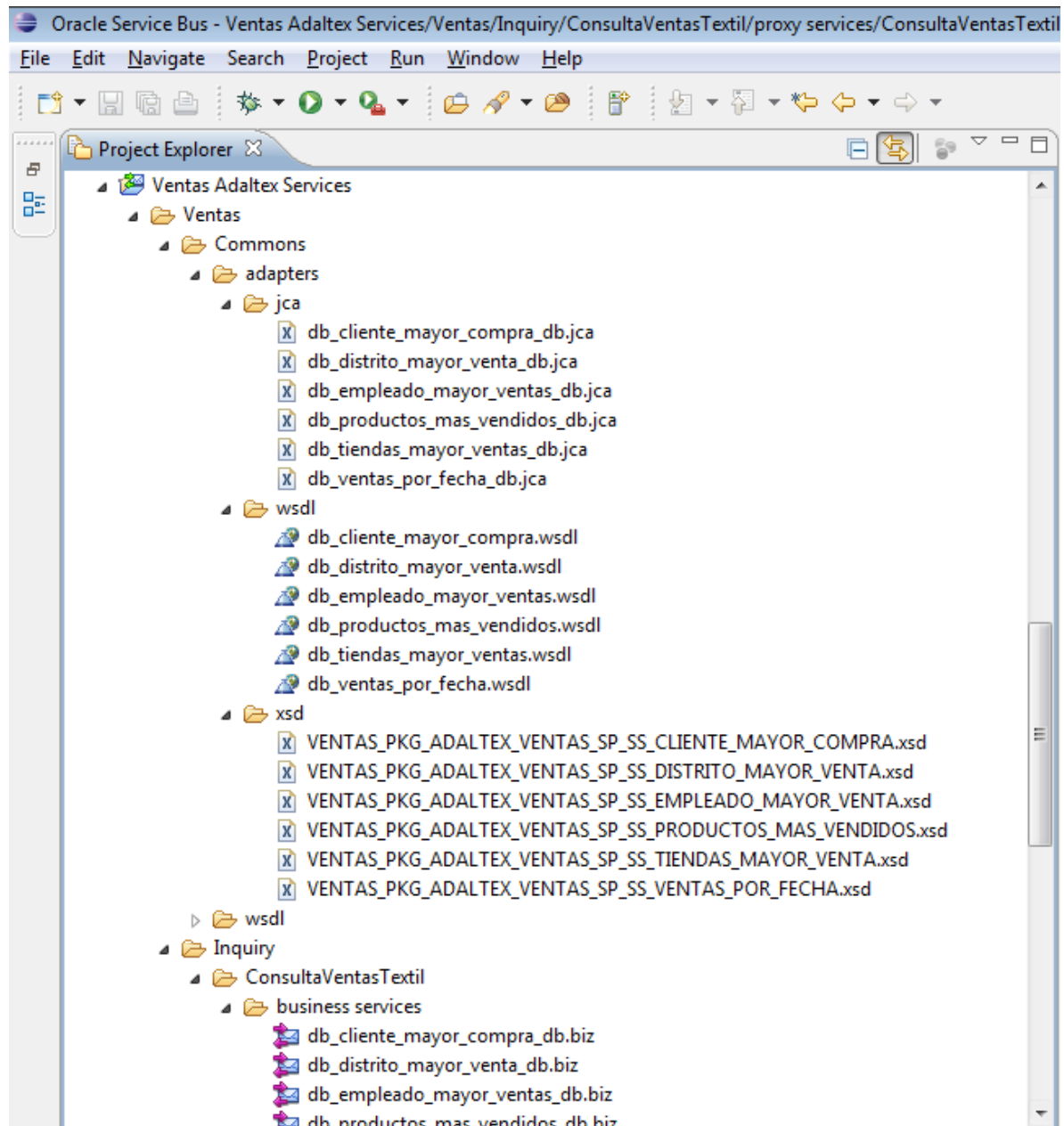
Figura 36. Diagrama Modelo de Datos

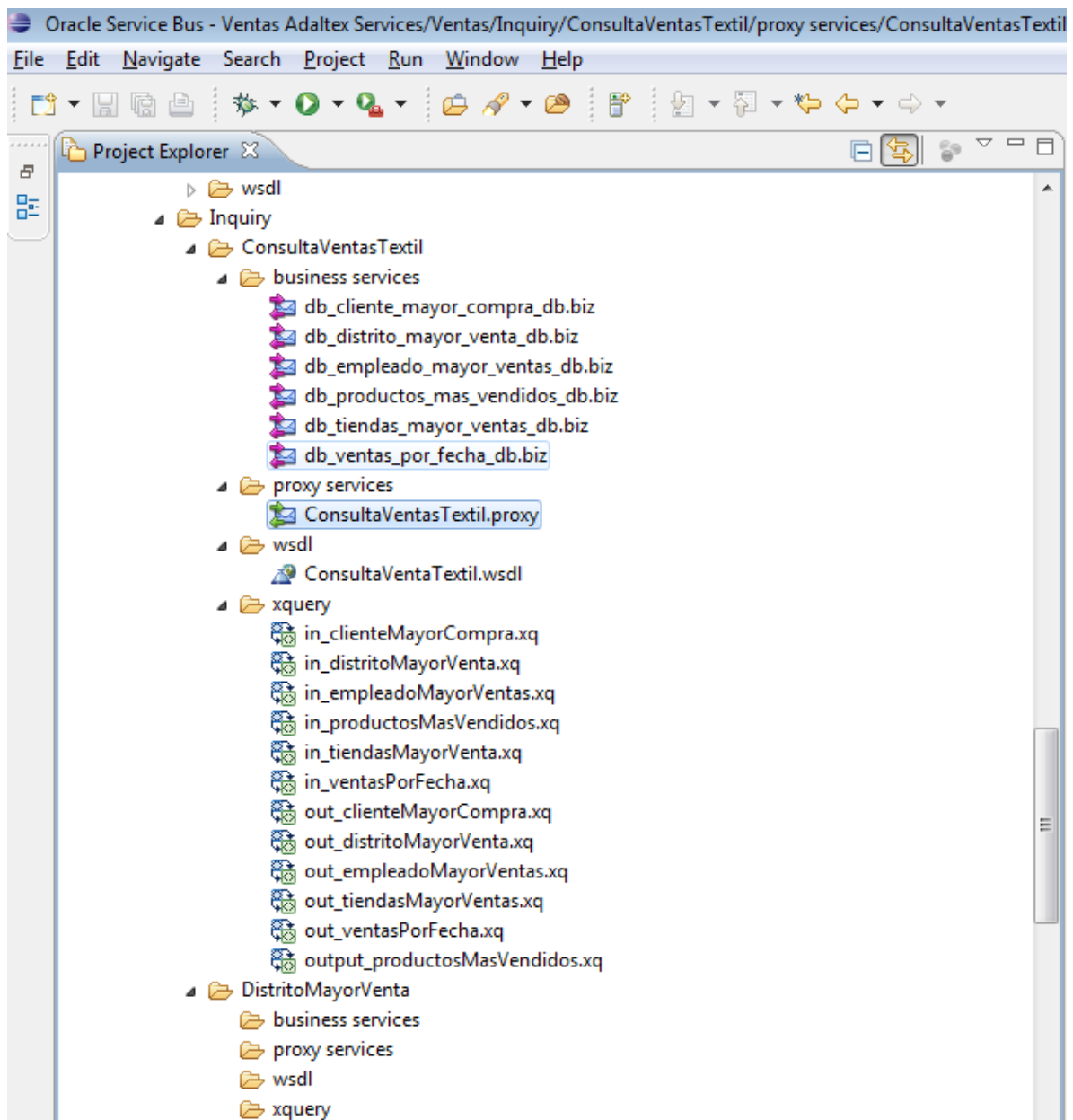


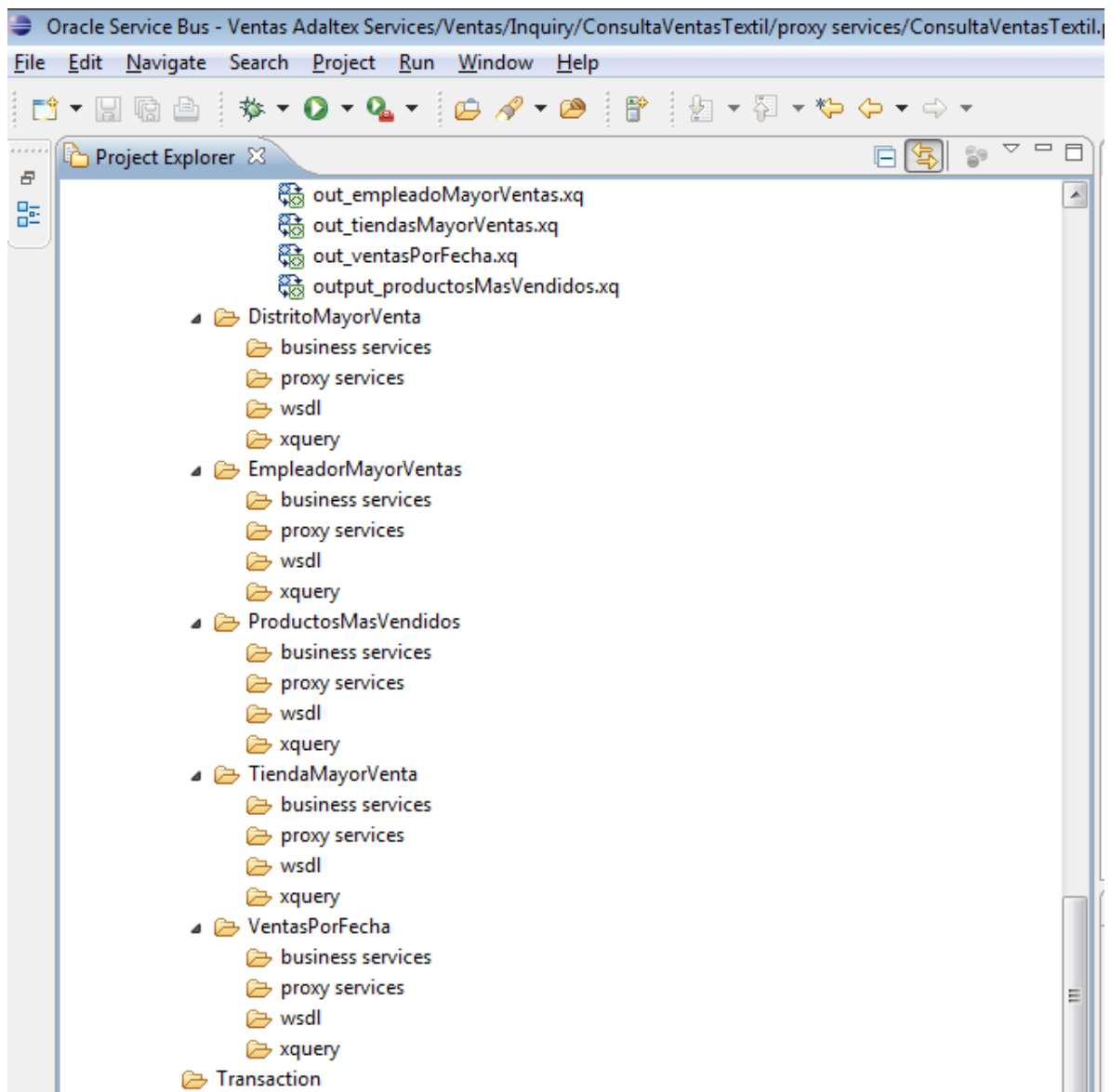
V APOORTE PRÁCTICO

5.1 DISEÑO DE LA SOLUCIÓN SOBRE LA ARQUITECTURA DEL ORACLE SERVICE BUS (OSB)

Estructura del Proyecto:







5.2 CÓDIGO FUENTE DE LOS SERVICIOS WEB MÁS IMPORTANTES:

- Web Service de consulta de los clientes que realizan más compras.

```
<?binding.jca db_cliente_mayor_compra_db.jca?>
<wsdl:definitions name="db_cliente_mayor_compra"
targetNamespace="http://xmlns.oracle.com/pcbpel/adapter/db/AdaltexProject
/ClienteMayorCompra/db_cliente_mayor_compra"
xmlns:db="http://xmlns.oracle.com/pcbpel/adapter/db/VENTAS/PKG_ADALTEX_VE
NTAS/SP_SS_CLIENTE_MAYOR_COMPRA/"
xmlns:tns="http://xmlns.oracle.com/pcbpel/adapter/db/AdaltexProject/Clien
teMayorCompra/db_cliente_mayor_compra"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:plt="http://schemas.xmlsoap.org/ws/2003/05/partner-link/">
  <plt:partnerLinkType name="db_cliente_mayor_compra_plt">
    <plt:role name="db_cliente_mayor_compra_role">
      <plt:portType name="tns:db_cliente_mayor_compra_ptt"/>
    </plt:role>
  </plt:partnerLinkType>
  <wsdl:types>
    <schema xmlns="http://www.w3.org/2001/XMLSchema">
      <import
namespace="http://xmlns.oracle.com/pcbpel/adapter/db/VENTAS/PKG_ADALTEX_V
ENTAS/SP_SS_CLIENTE_MAYOR_COMPRA/"
schemaLocation="../../../xsd/VENTAS_PKG_ADALTEX_VENTAS_SP_SS_CLIENTE_MAYOR_COMP
RA.xsd"/>
    </schema>
  </wsdl:types>
  <wsdl:message name="args_in_msg">
    <wsdl:part name="InputParameters" element="db:InputParameters"/>
  </wsdl:message>
  <wsdl:message name="args_out_msg">
    <wsdl:part name="OutputParameters"
element="db:OutputParameters"/>
  </wsdl:message>
  <wsdl:portType name="db_cliente_mayor_compra_ptt">
    <wsdl:operation name="db_cliente_mayor_compra">
      <wsdl:input message="tns:args_in_msg"/>
      <wsdl:output message="tns:args_out_msg"/>
    </wsdl:operation>
  </wsdl:portType>
</wsdl:definitions>
```

- **Web Service de consulta de los distritos dónde se realizan la mayor cantidad de ventas**

```
<?binding.jca db_distrito_mayor_venta_db.jca?>
<wsdl:definitions name="db_distrito_mayor_venta"
targetNamespace="http://xmlns.oracle.com/pcbpel/adapter/db/AdaltexProject
/DistritoMayorVenta/db_distrito_mayor_venta"
xmlns:db="http://xmlns.oracle.com/pcbpel/adapter/db/VENTAS/PKG_ADALTEX_VE
NTAS/SP_SS_DISTRITO_MAYOR_VENTA/"
xmlns:tns="http://xmlns.oracle.com/pcbpel/adapter/db/AdaltexProject/Distr
itoMayorVenta/db_distrito_mayor_venta"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:plt="http://schemas.xmlsoap.org/ws/2003/05/partner-link/">
  <plt:partnerLinkType name="db_distrito_mayor_venta_plt">
    <plt:role name="db_distrito_mayor_venta_role">
      <plt:portType name="tns:db_distrito_mayor_venta_ptt"/>
    </plt:role>
  </plt:partnerLinkType>
  <wsdl:types>
    <schema xmlns="http://www.w3.org/2001/XMLSchema">
      <import
namespace="http://xmlns.oracle.com/pcbpel/adapter/db/VENTAS/PKG_ADALTEX_V
ENTAS/SP_SS_DISTRITO_MAYOR_VENTA/"
schemaLocation="../../../xsd/VENTAS_PKG_ADALTEX_VENTAS_SP_SS_DISTRITO_MAYOR_VEN
TA.xsd"/>
    </schema>
  </wsdl:types>
  <wsdl:message name="args_in_msg">
    <wsdl:part name="InputParameters" element="db:InputParameters"/>
  </wsdl:message>
  <wsdl:message name="args_out_msg">
    <wsdl:part name="OutputParameters"
element="db:OutputParameters"/>
  </wsdl:message>
  <wsdl:portType name="db_distrito_mayor_venta_ptt">
    <wsdl:operation name="db_distrito_mayor_venta">
      <wsdl:input message="tns:args_in_msg"/>
      <wsdl:output message="tns:args_out_msg"/>
    </wsdl:operation>
  </wsdl:portType>
</wsdl:definitions>
```

- **Web Service de consulta de los empleados que realizan la mayor cantidad de ventas:**

```
<?binding.jca db_empleado_mayor_ventas_db.jca?>
<wsdl:definitions name="db_empleado_mayor_ventas"
targetNamespace="http://xmlns.oracle.com/pcbpel/adapter/db/AdaltexProject
/EmpleadoMayorVenta/db_empleado_mayor_ventas"
xmlns:db="http://xmlns.oracle.com/pcbpel/adapter/db/VENTAS/PKG_ADALTEX_VE
NTAS/SP_SS_EMPLEADO_MAYOR_VENTA/"
xmlns:tns="http://xmlns.oracle.com/pcbpel/adapter/db/AdaltexProject/Emple
adoMayorVenta/db_empleado_mayor_ventas"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:plt="http://schemas.xmlsoap.org/ws/2003/05/partner-link/">
  <plt:partnerLinkType name="db_empleado_mayor_ventas_plt">
    <plt:role name="db_empleado_mayor_ventas_role">
      <plt:portType name="tns:db_empleado_mayor_ventas_ptt"/>
    </plt:role>
  </plt:partnerLinkType>
  <wsdl:types>
    <schema xmlns="http://www.w3.org/2001/XMLSchema">
      <import
namespace="http://xmlns.oracle.com/pcbpel/adapter/db/VENTAS/PKG_ADALTEX_V
ENTAS/SP_SS_EMPLEADO_MAYOR_VENTA/"
schemaLocation="../../../xsd/VENTAS_PKG_ADALTEX_VENTAS_SP_SS_EMPLEADO_MAYOR_VEN
TA.xsd"/>
    </schema>
  </wsdl:types>
  <wsdl:message name="args_in_msg">
    <wsdl:part name="InputParameters" element="db:InputParameters"/>
  </wsdl:message>
  <wsdl:message name="args_out_msg">
    <wsdl:part name="OutputParameters"
element="db:OutputParameters"/>
  </wsdl:message>
  <wsdl:portType name="db_empleado_mayor_ventas_ptt">
    <wsdl:operation name="db_empleado_mayor_ventas">
      <wsdl:input message="tns:args_in_msg"/>
      <wsdl:output message="tns:args_out_msg"/>
    </wsdl:operation>
  </wsdl:portType>
</wsdl:definitions>
```

- **Web Service de consulta de los productos que registran mayor cantidad de ventas:**

```
<?binding.jca db_productos_mas_vendidos_db.jca?>
<wsdl:definitions name="db_productos_mas_vendidos"
targetNamespace="http://xmlns.oracle.com/pcbpel/adapter/db/AdaltexProject
/ProductosMasVendidos/db_productos_mas_vendidos"
xmlns:db="http://xmlns.oracle.com/pcbpel/adapter/db/VENTAS/PKG_ADALTEX_VE
NTAS/SP_SS_PRODUCTOS_MAS_VENDIDOS/"
xmlns:tns="http://xmlns.oracle.com/pcbpel/adapter/db/AdaltexProject/Produ
ctosMasVendidos/db_productos_mas_vendidos"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:plt="http://schemas.xmlsoap.org/ws/2003/05/partner-link/">
  <plt:partnerLinkType name="db_productos_mas_vendidos_plt">
    <plt:role name="db_productos_mas_vendidos_role">
      <plt:portType name="tns:db_productos_mas_vendidos_ptt"/>
    </plt:role>
  </plt:partnerLinkType>
  <wsdl:types>
    <schema xmlns="http://www.w3.org/2001/XMLSchema">
      <import
namespace="http://xmlns.oracle.com/pcbpel/adapter/db/VENTAS/PKG_ADALTEX_V
ENTAS/SP_SS_PRODUCTOS_MAS_VENDIDOS/"
schemaLocation="../../../xsd/VENTAS_PKG_ADALTEX_VENTAS_SP_SS_PRODUCTOS_MAS_VEND
IDOS.xsd"/>
    </schema>
  </wsdl:types>
  <wsdl:message name="args_in_msg">
    <wsdl:part name="InputParameters" element="db:InputParameters"/>
  </wsdl:message>
  <wsdl:message name="args_out_msg">
    <wsdl:part name="OutputParameters"
element="db:OutputParameters"/>
  </wsdl:message>
  <wsdl:portType name="db_productos_mas_vendidos_ptt">
    <wsdl:operation name="db_productos_mas_vendidos">
      <wsdl:input message="tns:args_in_msg"/>
      <wsdl:output message="tns:args_out_msg"/>
    </wsdl:operation>
  </wsdl:portType>
</wsdl:definitions>
```


- **Web Service de consulta de las tiendas que registran la mayor cantidad de ventas:**

```
<?binding.jca db_tiendas_mayor_ventas_db.jca?>
<wsdl:definitions name="db_tiendas_mayor_ventas"
targetNamespace="http://xmlns.oracle.com/pcbpel/adapter/db/AdaltexProject
/TiendasMayorVenta/db_tiendas_mayor_ventas"
xmlns:db="http://xmlns.oracle.com/pcbpel/adapter/db/VENTAS/PKG_ADALTEX_VE
NTAS/SP_SS_TIENDAS_MAYOR_VENTA/"
xmlns:tns="http://xmlns.oracle.com/pcbpel/adapter/db/AdaltexProject/Tiend
asMayorVenta/db_tiendas_mayor_ventas"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:plt="http://schemas.xmlsoap.org/ws/2003/05/partner-link/">
  <plt:partnerLinkType name="db_tiendas_mayor_ventas_plt">
    <plt:role name="db_tiendas_mayor_ventas_role">
      <plt:portType name="tns:db_tiendas_mayor_ventas_ptt"/>
    </plt:role>
  </plt:partnerLinkType>
  <wsdl:types>
    <schema xmlns="http://www.w3.org/2001/XMLSchema">
      <import
namespace="http://xmlns.oracle.com/pcbpel/adapter/db/VENTAS/PKG_ADALTEX_V
ENTAS/SP_SS_TIENDAS_MAYOR_VENTA/"
schemaLocation="../../../xsd/VENTAS_PKG_ADALTEX_VENTAS_SP_SS_TIENDAS_MAYOR_VENT
A.xsd"/>
    </schema>
  </wsdl:types>
  <wsdl:message name="args_in_msg">
    <wsdl:part name="InputParameters" element="db:InputParameters"/>
  </wsdl:message>
  <wsdl:message name="args_out_msg">
    <wsdl:part name="OutputParameters"
element="db:OutputParameters"/>
  </wsdl:message>
  <wsdl:portType name="db_tiendas_mayor_ventas_ptt">
    <wsdl:operation name="db_tiendas_mayor_ventas">
      <wsdl:input message="tns:args_in_msg"/>
      <wsdl:output message="tns:args_out_msg"/>
    </wsdl:operation>
  </wsdl:portType>
</wsdl:definitions>
```

- **Web Service de consulta de las ventas realizadas durante un determinado rango de fechas**

```
<?binding.jca db_ventas_por_fecha_db.jca?>
<wsdl:definitions name="db_ventas_por_fecha"
targetNamespace="http://xmlns.oracle.com/pcbpel/adapter/db/AdaltexProject
/VentasPorFecha/db_ventas_por_fecha"
xmlns:db="http://xmlns.oracle.com/pcbpel/adapter/db/VENTAS/PKG_ADALTEX_VE
NTAS/SP_SS_VENTAS_POR_FECHA/"
xmlns:tns="http://xmlns.oracle.com/pcbpel/adapter/db/AdaltexProject/Venta
sPorFecha/db_ventas_por_fecha"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:plt="http://schemas.xmlsoap.org/ws/2003/05/partner-link/">
  <plt:partnerLinkType name="db_ventas_por_fecha_plt">
    <plt:role name="db_ventas_por_fecha_role">
      <plt:portType name="tns:db_ventas_por_fecha_ptt"/>
    </plt:role>
  </plt:partnerLinkType>
  <wsdl:types>
    <schema xmlns="http://www.w3.org/2001/XMLSchema">
      <import
namespace="http://xmlns.oracle.com/pcbpel/adapter/db/VENTAS/PKG_ADALTEX_V
ENTAS/SP_SS_VENTAS_POR_FECHA/"
schemaLocation="../../../xsd/VENTAS_PKG_ADALTEX_VENTAS_SP_SS_VENTAS_POR_FECHA.x
sd"/>
    </schema>
  </wsdl:types>
  <wsdl:message name="args_in_msg">
    <wsdl:part name="InputParameters" element="db:InputParameters"/>
  </wsdl:message>
  <wsdl:message name="args_out_msg">
    <wsdl:part name="OutputParameters"
element="db:OutputParameters"/>
  </wsdl:message>
  <wsdl:portType name="db_ventas_por_fecha_ptt">
    <wsdl:operation name="db_ventas_por_fecha">
      <wsdl:input message="tns:args_in_msg"/>
      <wsdl:output message="tns:args_out_msg"/>
    </wsdl:operation>
  </wsdl:portType>
</wsdl:definitions>
```

- **Web Service Principal** quien realiza la orquestación de los servicios web relacionados con las ventas de Adaltex:

```
<?xml version="1.0" encoding="UTF-8" ?>
<definitions
  targetNamespace="http://adaltex.com.pe/eai/ConsultaVentaTextil"
  xmlns="http://schemas.xmlsoap.org/wsdl/"
  xmlns:tns="http://adaltex.com.pe/eai/ConsultaVentaTextil"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/"
  xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
  >
  <types>
    <xsd:schema
      targetNamespace="http://adaltex.com.pe/eai/ConsultaVentaTextil"
      elementFormDefault="qualified">
      <xsd:element name="consultarClienteMayorCompraRequest">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="idTransaccion" type="xsd:string"/>
            <xsd:element name="nombreAplicacion" type="xsd:string"/>
            <xsd:element name="ipAplicacion" type="xsd:string"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="consultarClienteMayorCompraResponse">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="outResultado" type="xsd:decimal"
              minOccurs="0"
              nillable="true"/>
            <xsd:element name="outMensaje" type="xsd:string"
              minOccurs="0"
              nillable="true"/>
            <xsd:element name="outClienteMayorCompra" type="tns:RowSet"
              minOccurs="0" nillable="true"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="consultarDistritoMayorVentaRequest">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="idTransaccion" type="xsd:string"/>
            <xsd:element name="nombreAplicacion" type="xsd:string"/>
            <xsd:element name="ipAplicacion" type="xsd:string"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="consultarDistritoMayorVentaResponse">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="outResultado" type="xsd:decimal"
              minOccurs="0"
              nillable="true"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:schema>
  </types>
</definitions>
```

```

        <xsd:element name="outMensaje" type="xsd:string"
minOccurs="0"
                nillable="true"/>
        <xsd:element name="outDistritoMayorVenta" type="tns:RowSet"
                minOccurs="0" nillable="true"/>
    </xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="consultarEmpleadoMayorVentaRequest">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="idTransaccion" type="xsd:string"/>
            <xsd:element name="nombreAplicacion" type="xsd:string"/>
            <xsd:element name="ipAplicacion" type="xsd:string"/>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:element name="consultarEmpleadoMayorVentaResponse">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="outResultado" type="xsd:decimal"
minOccurs="0"
                nillable="true"/>
            <xsd:element name="outMensaje" type="xsd:string"
minOccurs="0"
                nillable="true"/>
            <xsd:element name="outEmpleadoMayorVenta" type="tns:RowSet"
minOccurs="0" nillable="true"/>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:element name="consultarProductoMasVendidoRequest">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="idTransaccion" type="xsd:string"/>
            <xsd:element name="nombreAplicacion" type="xsd:string"/>
            <xsd:element name="ipAplicacion" type="xsd:string"/>
            <xsd:element name="numeroProductos" type="xsd:decimal"/>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:element name="consultarProductoMasVendidoResponse">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="outResultado" type="xsd:decimal"
minOccurs="0"
                nillable="true"/>
            <xsd:element name="outMensaje" type="xsd:string"
minOccurs="0"
                nillable="true"/>
            <xsd:element name="outProductoMasVendido" type="tns:RowSet"
minOccurs="0" nillable="true"/>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:element name="consultarTiendaMayorVentaRequest">
    <xsd:complexType>

```

```

        <xsd:sequence>
            <xsd:element name="idTransaccion" type="xsd:string"/>
            <xsd:element name="nombreAplicacion" type="xsd:string"/>
            <xsd:element name="ipAplicacion" type="xsd:string"/>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:element name="consultarTiendaMayorVentaResponse">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="outResultado" type="xsd:decimal"
minOccurs="0"
                                nillable="true"/>
            <xsd:element name="outMensaje" type="xsd:string"
minOccurs="0"
                                nillable="true"/>
            <xsd:element name="outTiendaMayorVenta" minOccurs="0"
nillable="true" type="tns:RowSet"/>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:element name="consultarVentasPorFechaRequest">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="idTransaccion" type="xsd:string"/>
            <xsd:element name="nombreAplicacion" type="xsd:string"/>
            <xsd:element name="ipAplicacion" type="xsd:string"/>
            <xsd:element name="fechaInicio" type="xsd:dateTime"/>
            <xsd:element name="fechaFin" type="xsd:dateTime"/>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:element name="consultarVentasPorFechaResponse">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="outResultado" type="xsd:decimal"
minOccurs="0"
                                nillable="true"/>
            <xsd:element name="outMensaje" type="xsd:string"
minOccurs="0"
                                nillable="true"/>
            <xsd:element name="outVentasPorFecha" type="tns:RowSet"
minOccurs="0" nillable="true"/>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:complexType name="RowSet">
    <xsd:sequence>
        <xsd:element name="Row" minOccurs="0" maxOccurs="unbounded">
            <xsd:complexType>
                <xsd:sequence>
                    <xsd:element name="Column" maxOccurs="unbounded"
nillable="true">
                        <xsd:complexType>
                            <xsd:simpleContent>
                                <xsd:extension base="xsd:string">

```

```

                                <xsd:attribute name="name"
type="xsd:string" use="required"/>
                                <xsd:attribute name="sqltype"
type="xsd:string" use="required"/>
                                </xsd:extension>
                                </xsd:simpleContent>
                                </xsd:complexType>
                                </xsd:element>
                                </xsd:sequence>
                                </xsd:complexType>
                                </xsd:element>
                                </xsd:sequence>
                                </xsd:complexType>
                                </xsd:schema>
</types>
<message name="consultarClienteMayorCompraRequest">
  <part name="parameters"
element="tns:consultarClienteMayorCompraRequest"/>
</message>
<message name="consultarClienteMayorCompraResponse">
  <part name="parameters"
element="tns:consultarClienteMayorCompraResponse"/>
</message>
<message name="consultarDistritoMayorVentaRequest">
  <part name="parameters"
element="tns:consultarDistritoMayorVentaRequest"/>
</message>
<message name="consultarDistritoMayorVentaResponse">
  <part name="parameters"
element="tns:consultarDistritoMayorVentaResponse"/>
</message>
<message name="consultarEmpleadoMayorVentaRequest">
  <part name="parameters"
element="tns:consultarEmpleadoMayorVentaRequest"/>
</message>
<message name="consultarEmpleadoMayorVentaResponse">
  <part name="parameters"
element="tns:consultarEmpleadoMayorVentaResponse"/>
</message>
<message name="consultarProductoMasVendidoRequest">
  <part name="parameters"
element="tns:consultarProductoMasVendidoRequest"/>
</message>
<message name="consultarProductoMasVendidoResponse">
  <part name="parameters"
element="tns:consultarProductoMasVendidoResponse"/>
</message>
<message name="consultarTiendaMayorVentaRequest">
  <part name="parameters"
element="tns:consultarTiendaMayorVentaRequest"/>
</message>
<message name="consultarTiendaMayorVentaResponse">
  <part name="parameters"
element="tns:consultarTiendaMayorVentaResponse"/>
</message>
<message name="consultarVentasPorFechaRequest">

```

```

    <part name="parameters"
element="tns:consultarVentasPorFechaRequest"/>
  </message>
  <message name="consultarVentasPorFechaResponse">
    <part name="parameters"
element="tns:consultarVentasPorFechaResponse"/>
  </message>
  <portType name="EbsConsultaVentas">
    <operation name="consultarClienteMayorCompra">
      <input message="tns:consultarClienteMayorCompraRequest"/>
      <output message="tns:consultarClienteMayorCompraResponse"/>
    </operation>
    <operation name="consultarDistritoMayorVenta">
      <input message="tns:consultarDistritoMayorVentaRequest"/>
      <output message="tns:consultarDistritoMayorVentaResponse"/>
    </operation>
    <operation name="consultarEmpleadoMayorVenta">
      <input message="tns:consultarEmpleadoMayorVentaRequest"/>
      <output message="tns:consultarEmpleadoMayorVentaResponse"/>
    </operation>
    <operation name="consultarProductoMasVendido">
      <input message="tns:consultarProductoMasVendidoRequest"/>
      <output message="tns:consultarProductoMasVendidoResponse"/>
    </operation>
    <operation name="consultarTiendaMayorVenta">
      <input message="tns:consultarTiendaMayorVentaRequest"/>
      <output message="tns:consultarTiendaMayorVentaResponse"/>
    </operation>
    <operation name="consultarVentasPorFecha">
      <input message="tns:consultarVentasPorFechaRequest"/>
      <output message="tns:consultarVentasPorFechaResponse"/>
    </operation>
  </portType>
  <binding name="EbsConsultaVentasSOAPBinding"
type="tns:EbsConsultaVentas">
    <soap:binding style="document"
      transport="http://schemas.xmlsoap.org/soap/http"/>
    <operation name="consultarClienteMayorCompra">
      <soap:operation style="document"

soapAction="http://adaltex.com.pe/eai/ConsultaVentaTextil/consultarClient
eMayorCompra"/>
      <input>
        <soap:body use="literal" parts="parameters"/>
      </input>
      <output>
        <soap:body use="literal" parts="parameters"/>
      </output>
    </operation>
    <operation name="consultarDistritoMayorVenta">
      <soap:operation style="document"

soapAction="http://adaltex.com.pe/eai/ConsultaVentaTextil/consultarDistri
toMayorVenta"/>
      <input>
        <soap:body use="literal" parts="parameters"/>
      </input>

```

```

        <output>
            <soap:body use="literal" parts="parameters"/>
        </output>
    </operation>
    <operation name="consultarEmpleadoMayorVenta">
        <soap:operation style="document"

soapAction="http://adaltex.com.pe/eai/ConsultaVentaTextil/consultarEmplea
doMayorVenta"/>
        <input>
            <soap:body use="literal" parts="parameters"/>
        </input>
        <output>
            <soap:body use="literal" parts="parameters"/>
        </output>
    </operation>
    <operation name="consultarProductoMasVendido">
        <soap:operation style="document"

soapAction="http://adaltex.com.pe/eai/ConsultaVentaTextil/consultarProduc
toMasVendido"/>
        <input>
            <soap:body use="literal" parts="parameters"/>
        </input>
        <output>
            <soap:body use="literal" parts="parameters"/>
        </output>
    </operation>
    <operation name="consultarTiendaMayorVenta">
        <soap:operation style="document"

soapAction="http://adaltex.com.pe/eai/ConsultaVentaTextil/consultarTienda
MayorVenta"/>
        <input>
            <soap:body use="literal" parts="parameters"/>
        </input>
        <output>
            <soap:body use="literal" parts="parameters"/>
        </output>
    </operation>
    <operation name="consultarVentasPorFecha">
        <soap:operation style="document"

soapAction="http://adaltex.com.pe/eai/ConsultaVentaTextil/consultarVentas
PorFecha"/>
        <input>
            <soap:body use="literal" parts="parameters"/>
        </input>
        <output>
            <soap:body use="literal" parts="parameters"/>
        </output>
    </operation>
</binding>
<service name="EbsConsultaVentasService">
    <port name="EbsConsultaVentasSoapPort"
        binding="tns:EbsConsultaVentasSOAPBinding">
        <soap:address location="http://localhost"/>
    </port>
</service>

```

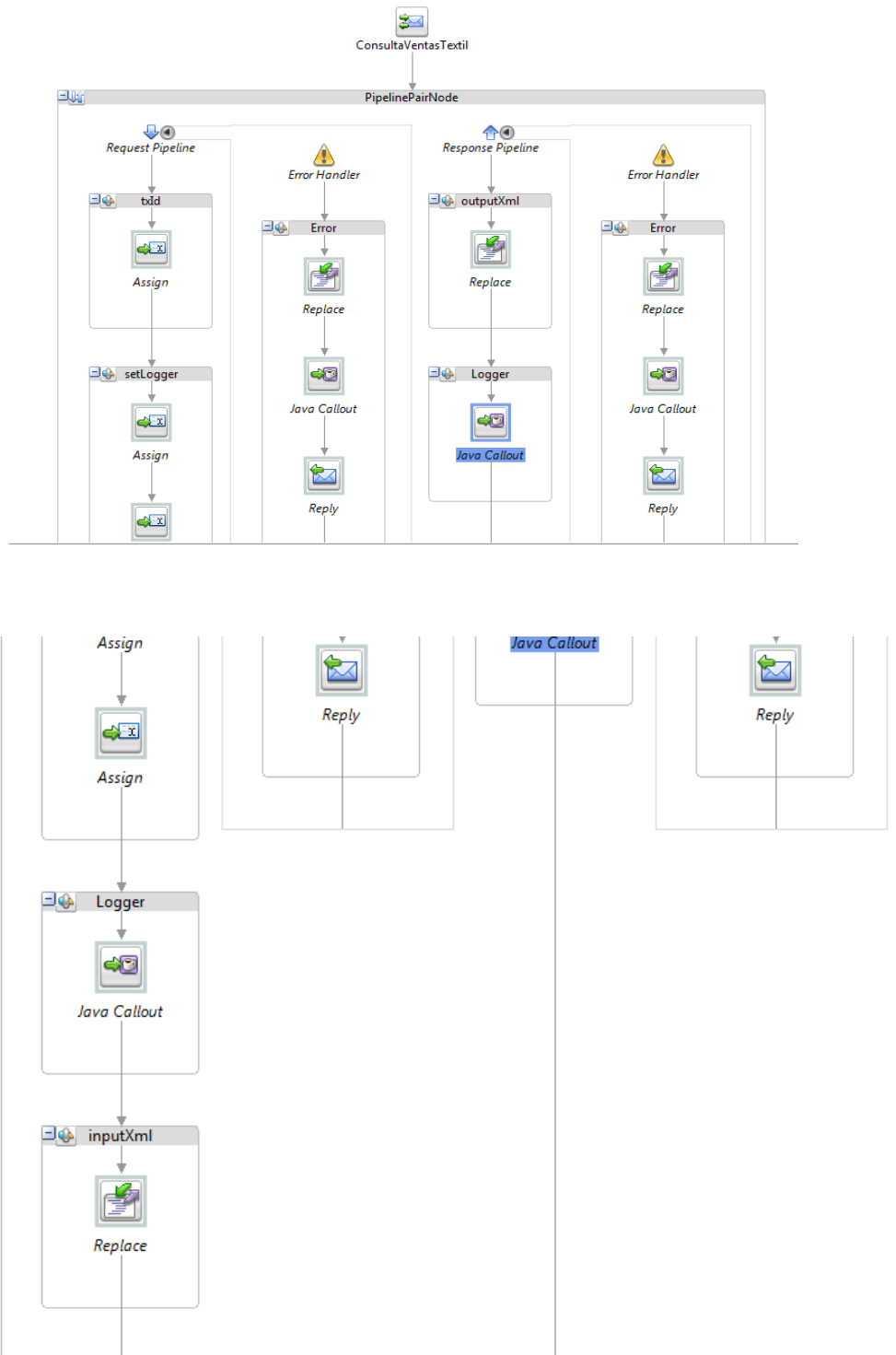


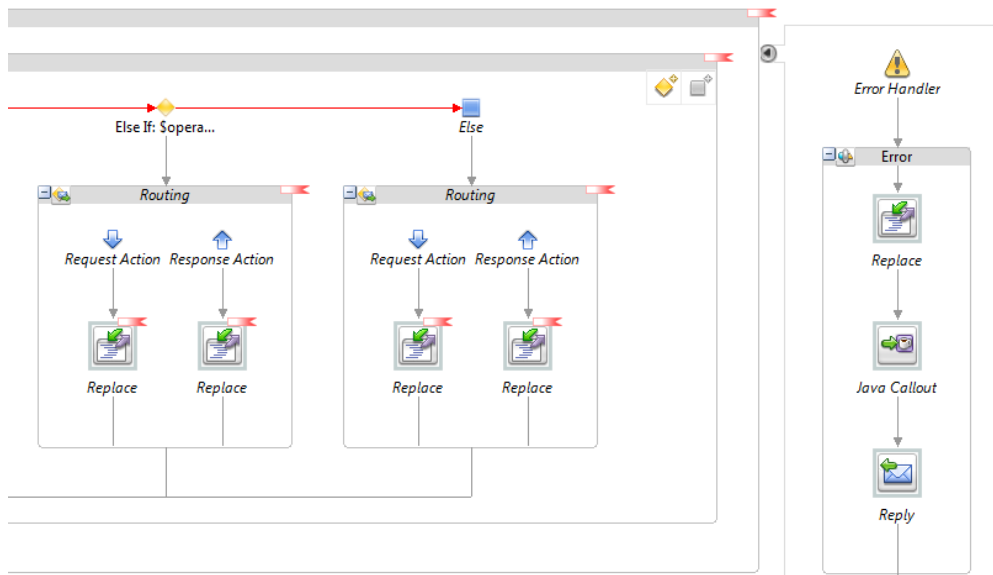
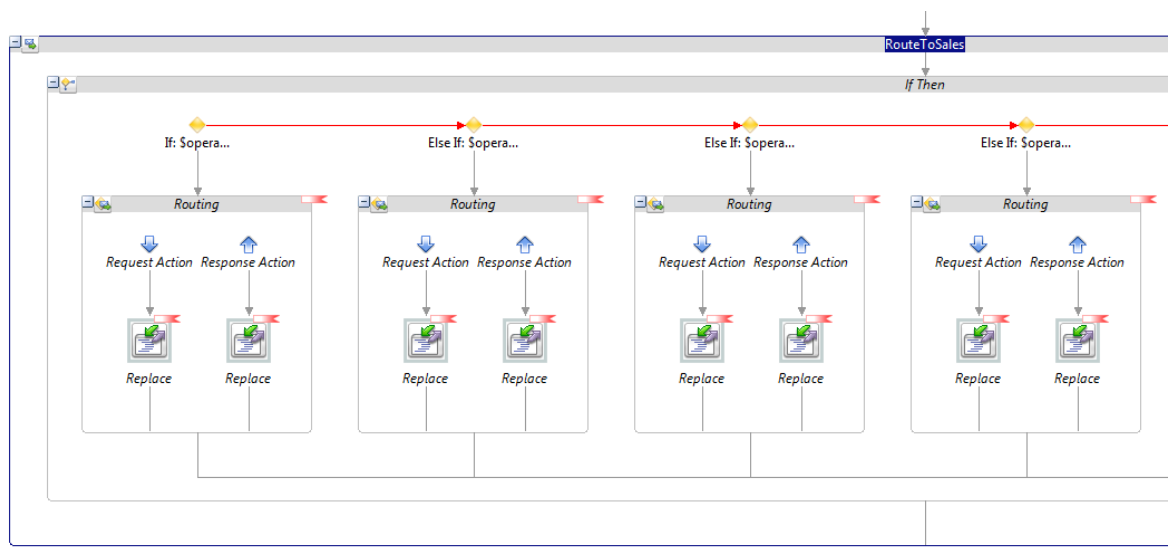
```

    </port>
  </service>
</definitions>

```

5.3 Diseño del flujo de comunicación de los servicios web a través del OSB.





VI. CONCLUSIONES

- Cuando se tienen sistemas integrados se reduce el tiempo de traslado de información generando ahorro en costos.
- Con una información inmediata entre los sistemas se reducen riesgos de error en transacciones pues además se asegura las correctas operaciones del sistema.
- La arquitectura de las tecnologías de la información es el componente esencial sobre el que se construyen relaciones provechosas.
- Una de las maneras de mantenerse a la altura de lo que demandan los clientes y de las exigencias del mercado consiste en llevar a cabo un repaso y una revisión continuos de los procesos empresariales.
- La integración afecta tan positivamente a sus clientes como a todos los que intervienen en la empresa.

BIBLIOGRAFÍA

Tesis

- [01]David Castro Salinas Interoperabilidad E Integración De Sistemas Informáticos De La Iglesia Católica En Chile
Tesis de grado, Ing. Informático, Universidad tecnológica Metropolitana Escuela de Informática, Chile, 2007
- [02]Elio San Cristóbal Ruiz Metodología, Estructura Y Desarrollo De Interfaces Intermedias Para La Conexión De Laboratorios Remotos Y Virtuales A Plataformas Educativas
Tesis Doctoral, Ingeniería en Informática, Universidad Nacional De Educación A Distancia, España, 2010
- [03]Jorge Cabrera Diaz Integración De Servicios Y Agentes De Usuario En La Recuperación De Información En Una Biblioteca Digital.
Tesis de Grado, Ingeniería en Sistemas Computacionales, Universidad de las Américas-Puebla, México, 2010
- [04]Gálvez Saldaña Carmen María Propuesta de Mejora del Proceso de Despacho para Incrementar la Satisfacción al cliente en una empresa comercializadora de productos de Acero.
Tesis para optar el título profesional de Ingeniero Industria, Universidad Peruana de Ciencias Aplicadas, 2008.
- [05]Paredes Gabriela Tesis de Ingeniería Comercial, Universidad Politécnica del Ejército, Quito, Ecuador, 2004.
- [06] Moreno Tesis Doctoral de P. Moreno. Una Aproximación Documental para la Creación e Integración de Juegos Digitales en Entornos Virtuales de Enseñanza. Universidad Complutense de Madrid, 2007.

Libros

- [07] Carrión Maroto Juan Estrategia: De la visión a la acción
2007, 461 Páginas.
- [08] Fernandez Gomez Eva Conocimientos y aplicaciones tecnológicas para la dirección
comercial.
2004, 247 Páginas.
- [09] Giner de la Fuente
Fernando Los sistemas de Información en la sociedad del conocimiento.
2004, 215 Páginas.
- [10] Jesús Sánchez Indicadores de Gestión Empresarial: De la Estrategia a los
Resultados.
Abril 2009
- [11] Sainz Andrés Ana Conozca el nuevo marketing: El valor de la información.
- [12] Salén Henrik Los secretos del merchandising activo o Cómo ser el número 1
en el punto de Venta

Revistas

- [13] EBSCO Journal Of Services Research.
Volumen 3, 13 Mayo 2004
- [14] Gartner Group Gartner's Hype Cycle
10 Junio 2004,
<http://www.floor.nl/ebiz/gartnershypecycle.htm>
- [15] Robert E. Kaplan y Mediciones que impulsan desempeño
David P. Norton
Harvard Business Review, 2009

Direcciones Electrónicas

- [16] Ramos Javier Samuel Tecnologías de información para e-business
<http://www.gestiopolis.com/canales2/gerencia/1/tisamuel.htm>
Junio 2004, 24 Julio 2012
- [17] Diez de Medina La Ingeniería de Negocios
Hernán
http://www.docirs.cl/ingenieria_negocios_hernandiezdemedina.htm
Marzo 2008, 15 Junio 2012
- [18] Herrera Cristhian Gestión de las Relaciones con el Cliente
<http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina>
- [19] McMaster
McMaster University. LMS Evaluation Information
<http://www.ltrc.mcmaster.ca/lmseval/>